

1. DAMP数据资产管理平台7.0LA用户手册	2
1.1 01 产品简介	3
1.2 02 产品安装指南	4
1.2.1 2.0 前提准备	5
1.2.2 2.1 初始化数据库	7
1.2.2.1 2.1.1 场景1: 元数据为基础应用	8
1.2.2.2 2.1.2 场景2: 数据标准为基础应用	10
1.2.2.3 2.1.3 场景3: 数据质量为基础应用	12
1.2.3 2.2 元数据管理平台安装部署	15
1.2.3.1 2.2.1 后端DGS服务部署	16
1.2.3.2 2.2.2 元数据采集器部署	18
1.2.4 2.3 数据资产目录安装部署	19
1.2.4.1 2.3.1 在线页面配置部署	20
1.2.4.2 2.3.2 数据服务部署	22
1.2.4.3 2.3.3 后端Server部署	25
1.2.5 2.4 数据标准管理平台安装部署	31
1.2.5.1 2.4.1 后端DSM服务部署	32
1.2.5.2 2.4.2 数据标准采集器部署	36
1.2.6 2.5 数据质量管理平台安装部署	37
1.2.6.1 2.5.1 检核执行器部署	38
1.2.6.2 2.5.2 调度服务部署	44
1.2.6.3 2.5.3 后端Server部署	48
1.2.7 2.6 Nginx配置	56
1.2.8 2.7 部署后修改参数	60
1.3 03 快速入门指南	65
1.3.1 3.1 元数据采集	66
1.3.2 3.2 资源同步、注册	67
1.3.3 3.3 资产数据质量检核	68
1.3.4 3.4 数据质量检核	70
1.3.5 3.5 服务发布	71
1.3.6 3.6 数据标准快速入门	72
1.4 04 前端显示配置	73
1.5 05 用户操作手册	75
1.6 06 FAQ	76
1.6.1 1. 元数据或数据标准server启动报错: Error creating bean with name 'dgsRemoteRpc' defined in class path resource [spring/corebean/context-adapter-mm-server.xml]	77

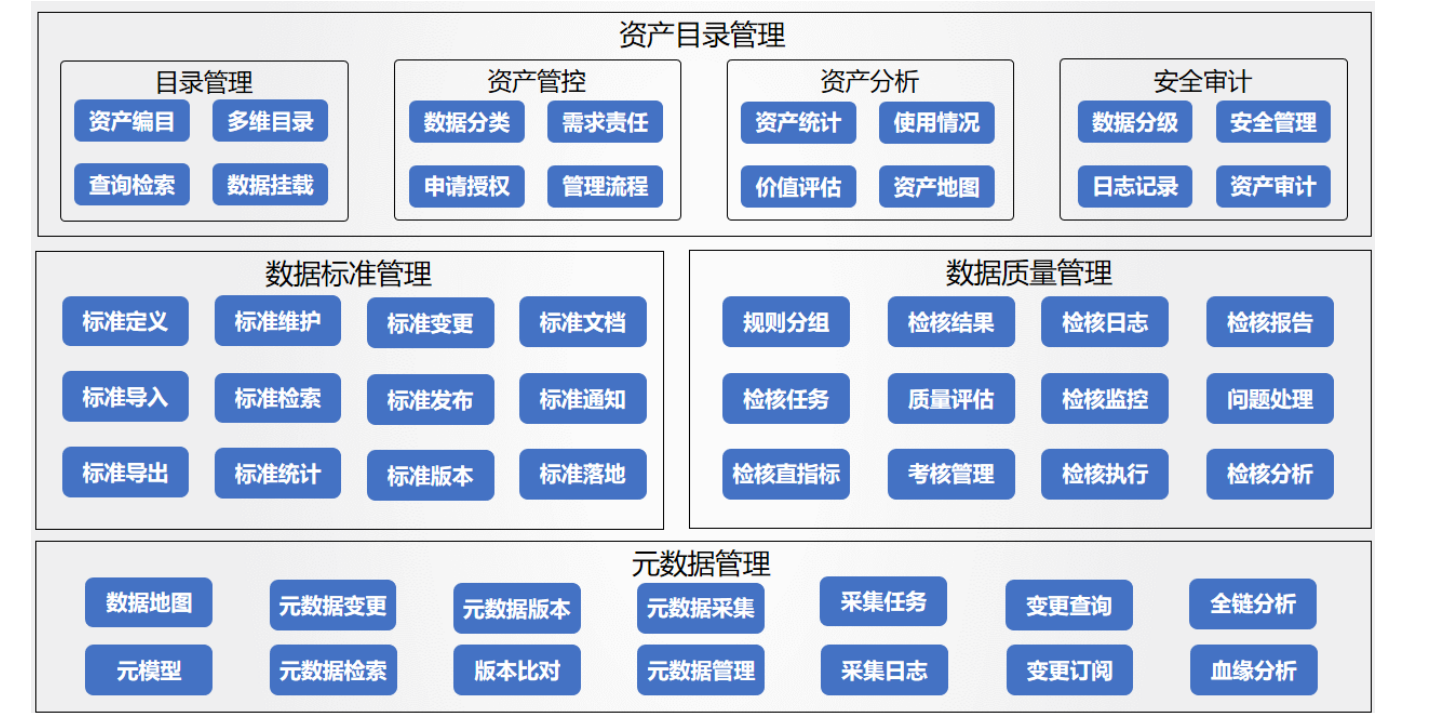
DAMP数据资产管理平台7.0LA用户手册

01 产品简介

普元数据资产管理平台(Primeton DAMP™) 定位于统筹管理企业数据资产、形成标准化、一致化的数据架构，为企业数字化转型下的业务变革与IT架构重塑提供数据基础。

基于普元数据资产管理平台，企业可以通过基础数据、业务数据、统计分析数据等的梳理与排摸，形成包含业务与技术要素的数据资产，在数据资产平台上落地与展现；通过数据标准规划与设计，优化数据模型架构，全面提升数据质量，为企业交付可靠数据；建立数据共享应用管理过程，提升企业数据管理能力，激发数据资源要素潜力，突破信息壁垒与业务藩篱，推动数据资产开发利用。

普元数据资产管理平台是具备数据资产全要素管理能力的企业级数据资产管理平台。它包含元数据管理、资产目录管理、数据标准管理、数据质量管理等能力。



元数据管理：提供IT全域元数据的模型定义、采集、存储、关联、版本、变更、发布等全过程管理能力，将元数据的业务要素、技术要素和管理要素形成关联，并基于元数据脉络提供数据地图、链路分析、关联度分析推荐等应用能力。

资产目录管理：提供一站式的数据资产分级分类、编目发布、展示检索、共享对接的整合门户，数据资产消费者、提供者、管理者可在统一的平台上，完成数据提供挂接、数据资产管理与数据共享获取。

数据标准管理：提供从标准设计制订、标准审核发布、标准落地映射的全过程管理，通过数据标准与元数据的连接映射，更好的驱动数据架构优化与数据质量提升。

数据质量管理：提供数据质量规则定义与检核执行能力，可适配不同的数据存储系统，可灵活自由的配置规则逻辑，并对检核结果建立流程化、规范化的处理机制，对数据质量问题反馈进行记录和考核。

02 产品安装指南

本指南中以元数据管理系统为基础应用，将数据资产目录、数据标准、数据质量集成到数据资产管理平台，并配置数据资产目录为登录的默认应用。

说明



选择不同应用作为基础应用，选择的初始化脚本会有不同，本示例以元数据管理系统为基础应用。

环境要求

项目	说明
CPU	主频2.0GHz以上
内存	8GB以上
硬盘	安装目录空间：20GB以上
操作系统	Windows 7/10、WindowsServer2008、CentOS7.2
数据库	MySQL 5.7.20
浏览器	Chrome 70及以上

标准产品 需要的组件及相关资源版本如下所示，请从官网下载数据库相关驱动程序文件。

资源名称	版本号	说明
JDK	v1.8.0_172	OracleJDK
Mysql 数据库驱动	v5.7.20	mysql-connector-java-5.1.40.jar

2.0 前提准备

本章节介绍部署集成环境需要使用的第三方软件的安装配置，标准介质中不提供第三方软件包，请参考下文中的版本按需下载。

因Redis、Solr、Nginx支持多种平台且安装方式有多种，本示例只介绍CentOS下的安装配置。

Redis 源码安装

1. 下载redis-5.0.12.tar.gz，上传到Linux服务器，并解压（redis下载地址：<https://download.redis.io/releases/>）。

```
#wget http://download.redis.io/releases/redis-5.0.12.tar.gz
#tar -zxvf redis-5.0.12.tar.gz -C /opt/
```

2. 预编译（实际上是检查编译环境的过程）。

进入目录：`cd /opt/redis-5.0.12/deps/jemalloc`

执行预编译 `./configure`

在预编译的过程中，会检测安装redis所需的相关依赖，依次安装即可(缺少c编译环境 `yum -y install gcc-c++`)。

说明

预编译不是必须的步骤，它只是在检查编译过程中需要的环境是否满足。

通常源码包中，都有一个可执行的configure脚本，这个脚本执行预编译的脚本。但是有一些源码包中，没有该文件，可以省略预编译步骤。

3. 编译
`#cd /opt/redis-5.0.12/src`
4. 安装
`#make`
`#cd /opt/redis-5.0.12/src`
`#make install`
5. 配置
修改redis.conf文件。
`# vim /opt/redis-5.0.12/redis.conf`

```
##### SECURITY #####
# Require clients to issue AUTH <PASSWORD> before processing any other
# commands. This might be useful in environments in which you do not trust
# others with access to the host running redis-server.
#
# This should stay commented out for backward compatibility and because most
# people do not need auth (e.g. they run their own servers).
#
# Warning: since Redis is pretty fast an outside user can try up to
# 150k passwords per second against a good box. This means that you should
# use a very strong password otherwise it will be very easy to break.
#
requirepass foobared    去掉注释，修改为自己的password
# Command renaming.
#
# It is possible to change the name of dangerous commands in a shared
# environment. For instance the CONFIG command may be renamed into something
# hard to guess so that it will still be available for internal-use tools
# but not available for general clients.
#
# Example:
#
# bind 192.168.1.100 10.0.0.1
# bind 127.0.0.1 ::1
#
# --- WARNING --- If the computer running Redis is directly exposed to the
# internet, binding to all the interfaces is dangerous and will expose the
# instance to everybody on the internet. So by default we uncomment the
# following bind directive, that will force Redis to listen only into
# the IPv4 loopback interface address (this means Redis will be able to
# accept connections only from clients running into the same computer it
# is running).
#
# IF YOU ARE SURE YOU WANT YOUR INSTANCE TO LISTEN TO ALL THE INTERFACES
# JUST COMMENT THE FOLLOWING LINE.
# ~~~~~
bind 127.0.0.1    #指定redis只接收来自于该IP地址的请求，如果不进行设置，那么将处理所有请求，在生产环境中最好设置该项
# Protected mode is a layer of security protection, in order to avoid that
# Redis instances left open on the internet are accessed and exploited.
#
# When protected mode is on and if:
#
# 1) The server is not binding explicitly to a set of addresses using the
# "bind" directive.
# 2) No password is configured.
#
# The server only accepts connections from clients connecting from the
# IPv4 and IPv6 loopback addresses 127.0.0.1 and ::1, and from Unix domain
# sockets.
#
:bind
```

其他修改可根据环境进行配置。

6. 启动
`# cd /opt/redis-5.0.12/src`
`# ./redis-server ../redis.conf`

7. 测试是否启动成功

```
#cd /opt/redis-5.0.12/src
#./redis-cli -a "密码"
```

```
[root@node3 src]# ./redis-cli -a "primeton000000"
Warning: Using a password with '-a' or '-u' option on the command line interface may not be safe.
127.0.0.1:6379> set aaa 1234
OK
127.0.0.1:6379> get aaa
"1234"
127.0.0.1:6379> exit
```

Solr安装配置

1. 下载Solr介质。（Solr下载地址：<http://archive.apache.org/dist/lucene/solr/>）

验证的版本为：solr-8.6.3.tar.gz。

2. 安装

解压下载的介质：# tar -zxvf solr-8.6.3.tar.gz

3. 配置

a. 将 **dgp.zip** 压缩包中的dgp文件夹拷贝到 {solr安装目录}/server/solr目录下，如下图。

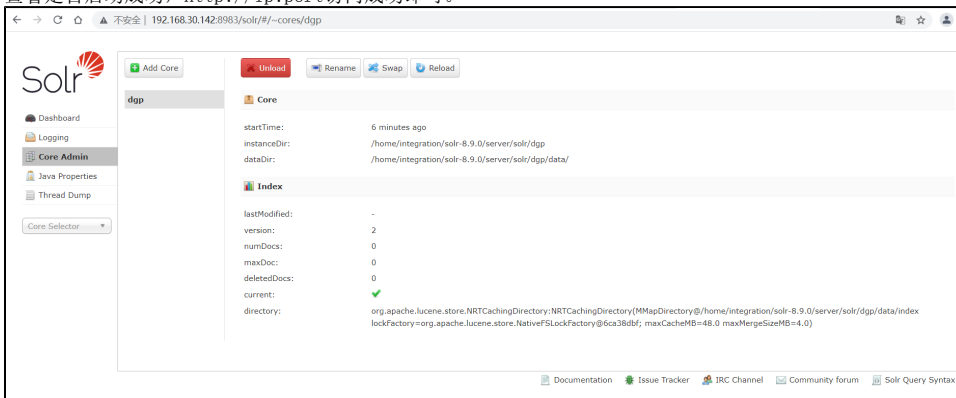
```
[root@primeton solr]# ll
total 12
drwxr-xr-x. 4 root root 55 Jun 11 04:23 configsets
drwxr-xr-x. 4 root root 50 Aug 5 18:02 dgp
drwxr-xr-x. 2 root root 6 Aug 5 18:04 filestore
-rw-r--r--. 1 root root 3018 Jun 11 04:23 README.txt
-rw-r--r--. 1 root root 2427 Jun 11 04:23 solr.xml
drwxr-xr-x. 2 root root 6 Aug 5 18:04 userfiles
-rw-r--r--. 1 root root 1048 Jun 11 04:23 zoo.cfg
[root@primeton solr]# pwd
/home/integration/solr-8.9.0/server/solr
[root@primeton solr]#
```

b. 将 **dgp.zip** 压缩包中的ik-analyzer-8.3.0.jar和ik-analyzer-solr7-7.x.jar 拷贝到 {solr安装目录}/server/solr-webapp/webapp/WEB-INF/lib目录下，如下图。

```
[root@primeton lib]# pwd
/home/integration/solr-8.9.0/server/solr-webapp/webapp/WEB-INF/lib
[root@primeton lib]# ll ik*
-rw-r--r--. 1 root root 9686058 Aug 5 18:03 ik-analyzer-8.3.0.jar
-rw-r--r--. 1 root root 1188360 Aug 5 18:03 ik-analyzer-solr7-7.x.jar
[root@primeton lib]#
```

4. 启动Solr

进入 {Solr安装目录}/bin目录下，执行 ./solr start -force （默认端口是8983）。

5. 查看是否启动成功，<http://ip:port>访问成功即可。

Nginx安装配置

1. 下载Nginx介质。（Nginx下载地址：<http://nginx.org/download/>）

验证的版本为：nginx/1.14.2

2. 安装

解压下载的介质：# tar -zxvf nginx-1.14.2.tar.gz

3. 启动

nginx

2.1 初始化数据库

脚本说明

db_scripts.zip脚本包中包含不同安装模式下的脚本文件，需要根据当前环境实际情况选择对应的脚本。此处以元数据的脚本为例进行说明。

元数据管理平台脚本包解压后目录如下：

db_scripts	
名称	
 independent	
 integration_non_dsm	
 integration_with_dsm	

independent：独立安装或者该应用作为主应用时使用。（主应用：集成环境下最先部署的应用，其他应用往主应用上进行集成。）

integration_non_xxx：该应用作为集成应用时，当前集成环境下没有xxx应用时使用该目录下脚本。（如图中integration_non_dsm：将元数据作为集成应用部署时，若实际集成环境中未部署数据标准平台，则使用该目录下脚本初始化元数据应用。）

integration_with_xxx：该应用作为集成应用时，当前集成环境下已有xxx应用时使用该目录下脚本。（如图中integration_with_dsm：将元数据作为集成应用部署时，若实际集成环境中已部署数据标准平台，则使用该目录下脚本初始化元数据应用。）

说明

 其他应用（数据资产、数据质量、数据标准）的脚本包也有类似的分类目录，参考上述示例说明。

初始化前数据库准备

1. Mysql数据库执行初始化脚本请用数据库客户端工具。
2. mysql配置文件my.cnf 中添加：log_bin_trust_function_creators=1 或者在数据库命令行模式下，使用root用户执行：set global log_bin_trust_function_creators=TRUE；否则创建函数的语句执行不成功。
3. 如果安装的是Mysql15.7版本，需要修改数据库 my.cnf 文件的 sql_mode 参数，修改如下所示。

```
# sql_mode=ONLY_FULL_GROUP_BY,.....  
# ONLY_FULL_GROUP_BY  
sql_mode=STRICT_TRANS_TABLES,ERROR_FOR_DIVISION_BY_ZERO,NO_AUTO_CREATE_USER,NO_ENGINE_SUBSTITUTION
```

2.1.1 场景1：元数据为基础应用

注意

- 1. 初始化数据库需要指定基础应用，其他应用集成到基础应用。因此指定的基础应用不同时，使用的脚本会有不同。**所有脚本均在同一数据库下执行。**
- 2. 本示例以**元数据管理平台为基础应用**，依次集成数据标准、数据资产目录、数据质量。
- 3. 若实际环境中仅部署部分应用，请根据实际情况选择对应的初始化脚本（脚本说明参考2.1 初始化数据库）。例如不需要部署数据标准，则可以跳过数据标准脚本初始化。

元数据脚本初始化

将元数据脚本 db_scripts.zip解压后，依次执行db_scripts\independent\mysql目录下所有子目录（create、initdata、sp）中的脚本。

说明

此示例中元数据作为基础应用，故元数据使用independent目录下的脚本。

MetaCube75GA > integration > db_scripts > independent > mysql >			
名称	修改日期	类型	大小
create	2021/7/27 9:33	文件夹	
initdata	2021/7/27 9:33	文件夹	
sp	2021/7/27 9:33	文件夹	

数据标准脚本初始化

将数据标准脚本 db_scripts.zip解压后，依次执行db_scripts\integration_with_metacube\mysql\initdata目录下所有脚本。

说明

此示例中数据标准作为集成应用和元数据（基础应用）做集成时，因当前集成环境中已部署元数据，故使用integration_with_metacube目录下的脚本。

> 标准db_scripts > integration_with_metacube > mysql > initdata			
名称	修改日期	类型	大小
00_coframe_data.sql	2021/7/27 9:37	DbVisualizer SQ...	46 KB
01_db_sys_init_data.sql	2021/7/27 9:37	DbVisualizer SQ...	5 KB

数据资产目录脚本初始化

将数据资产目录脚本 db_scripts.zip解压后，依次执行db_scripts\mysql\integration_non_dqms目录下所有子目录（create、initdata、sp）中的脚本。

说明

此示例中数据资产目录作为集成应用和元数据（基础应用）、数据标准做集成时，因当前集成环境中未部署数据质量（dqms），故使用 integration_non_dqms 目录下的脚本。

egration\资产目录dbscripts\mysql\integration_non_dqms				
名称	修改日期	类型	大小	
create	2021/7/27 10:00	文件夹		
initdata	2021/7/27 10:00	文件夹		
sp	2021/7/27 10:00	文件夹		

说明

1. 若执行脚本时出现如下报错，需切换超级管理员权限用户（root）执行脚本。

```
mysql> set global log_bin_trust_function_creators=TRUE;
ERROR 1227 (42000): Access denied; you need (at least one of) the SUPER privilege(s) for this operation
```

数据质量脚本初始化

将数据质量脚本 db_scripts.zip解压后，依次执行dbscripts\mysql\integration_with_damp目录下所有子目录（create、initdata、sp）中的脚本。

说明

此示例中数据质量作为集成应用和元数据（基础应用）、数据标准、数据资产目录做集成时，因当前集成环境中已部署数据资产目录（damp），故使用 integration_with_damp 目录下的脚本。

egration\质量dbscripts\mysql\integration_with_damp				
名称	修改日期	类型	大小	
create	2021/7/27 10:00	文件夹		
initdata	2021/7/27 10:00	文件夹		
sp	2021/7/27 10:00	文件夹		

2.1.2 场景2：数据标准为基础应用

注意

- 1. 初始化数据库需要指定基础应用，其他应用集成到基础应用。因此指定的基础应用不同时，使用的脚本会有不同。**所有脚本均在同一数据库下执行。**
- 2. 本示例以**数据标准管理平台为基础应用**，依次集成元数据、数据质量、数据资产目录。
- 3. 若实际环境中仅部署部分应用，请根据实际情况选择对应的初始化脚本（脚本说明参考2.1 初始化数据库）。

数据标准脚本初始化

将数据标准脚本 db_scripts.zip解压后，依次执行db_scripts\independent\mysql目录下所有子目录（coframe_option、create、initdata、sp）中的脚本。

说明

此示例中数据标准作为基础应用，故数据标准使用independent目录下的脚本。

75 > dsm7LA > db_scripts > indendpent > mysql >			搜索"mysql"
名称	修改日期	类型	
coframe_option	2021/7/27 9:37	文件夹	
create	2021/7/27 9:37	文件夹	
initdata	2021/7/27 9:37	文件夹	
sp	2021/7/27 9:37	文件夹	

元数据脚本初始化

将元数据脚本 db_scripts.zip解压后，依次执行db_scripts\integration_with_dsm\mysql\initdata目录下所有脚本。

说明

此示例中元数据作为集成应用和数据标准（基础应用）做集成时，因当前集成环境中已部署数据标准，故使用integration_with_dsm目录下的脚本。

dgs75 > db_scripts > integration_with_dsm > mysql > initdata			搜索"initdata"
名称	修改日期	类型	
00_db_coframe_init_data.sql	2021/7/27 9:33	SQL 文件	

数据质量脚本初始化

将数据质量脚本 db_scripts.zip解压后，依次执行dbscripts\mysql\integration_non_damp目录下所有子目录（create、initdata、sp）中的脚本。

说明

此示例中数据质量作为集成应用和数据标准（基础应用）、元数据做集成时，因当前集成环境中未部署数据资产目录（damp），故使用 integration_non_damp 目录下的脚本。

« dbscripts > mysql > integration_non_damp				搜索"integra
	名称	修改日期	类型	
	create	2021/7/27 16:15	文件夹	
	initdata	2021/7/27 16:15	文件夹	
	sp	2021/7/27 16:15	文件夹	

数据资产目录脚本初始化

将数据资产目录脚本 db_scripts.zip解压后，依次执行dbscripts\mysql\integration_with_dqms目录下所有子目录（initdata）中的脚本。

说明

此示例中数据资产目录作为集成应用和数据标准（基础应用）、元数据、数据质量做集成时，因当前集成环境中已部署数据质量（dqms），故使用 integration_with_dqms 目录下的脚本。

« dbscripts > mysql > integration_with_dqms				搜
	名称	修改日期		
	initdata	2021/7/27 16:14		

说明

1. 若执行脚本时出现如下报错，需切换超级管理员权限用户（root）执行脚本。

```
mysql> set global log_bin_trust_function_creators=TRUE;
ERROR 1227 (42000): Access denied; you need (at least one of) the SUPER privilege(s) for this operation
```

2.1.3 场景3：数据质量为基础应用

注意

1. 初始化数据库需要指定基础应用，其他应用集成到基础应用。因此指定的基础应用不同时，使用的脚本会有不同。**所有脚本均在同一数据库下执行。**
2. 本示例以**数据质量管理平台为基础应用**，依次集成数据资产目录、元数据、数据标准。
3. 若实际环境中仅部署部分应用，请根据实际情况选择对应的初始化脚本（脚本说明参考2.1 初始化数据库）。

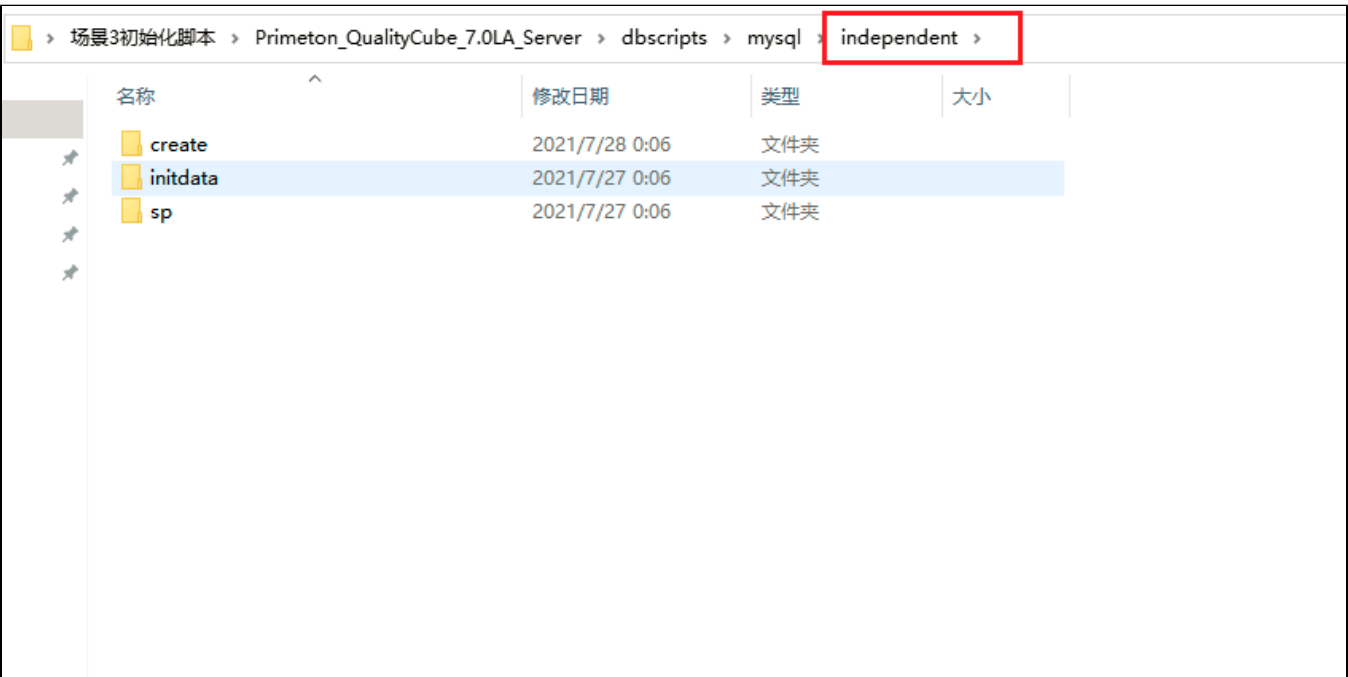
数据质量脚本初始化

. 将数据质量脚本 db_scripts.zip解压后，依次执行dbscripts\mysql\independent目录下所有子目录（create、initdata、sp）中的脚本。

说明

1. 此示例中数据质量作为基础应用，故数据质量使用independent目录下的脚本。
2. 若执行脚本时出现如下报错，需切换超级管理员权限用户（root）执行脚本。

```
mysql> set global log_bin_trust_function_creators=TRUE;
ERROR 1227 (42000): Access denied; you need (at least one of) the SUPER privilege(s) for this operation
```

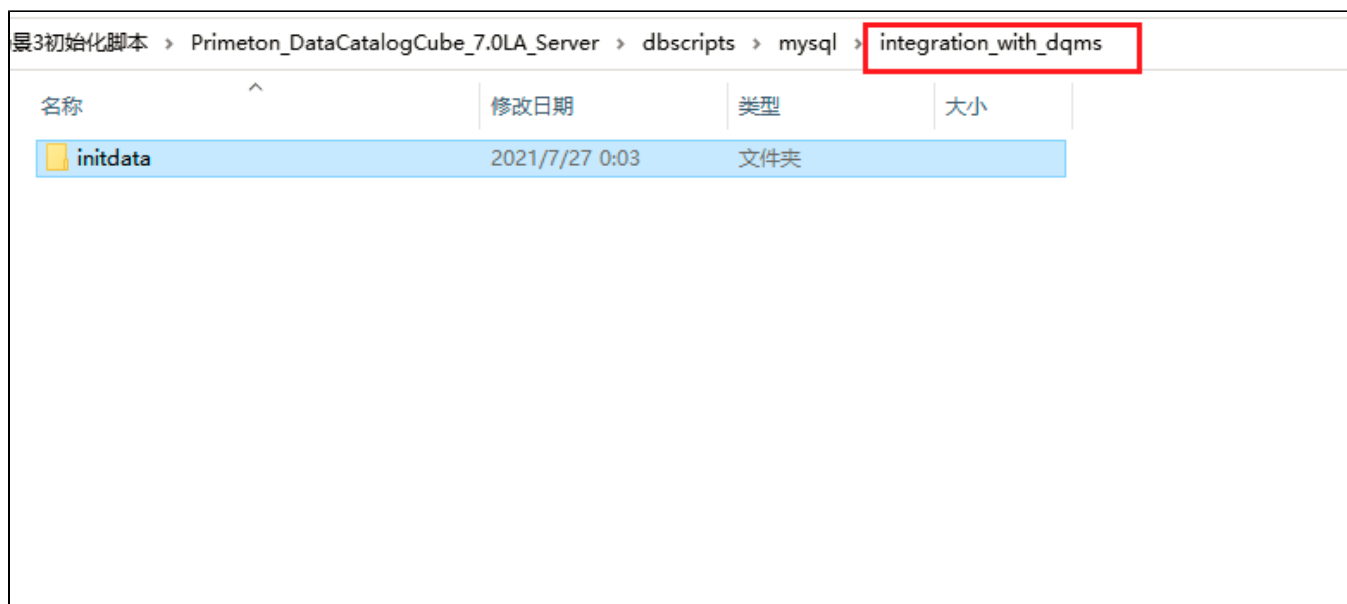


数据资产目录脚本初始化

将数据资产目录脚本 db_scripts.zip解压后，依次执行dbscripts\mysql\integration_with_dqms目录下所有子目录（initdata）中的脚本。

注意

此示例中数据资产目录作为集成应用和数据质量（基础应用）做集成时，因当前集成环境中已部署数据质量（dqms），故使用integration_with_dqms目录下的脚本。

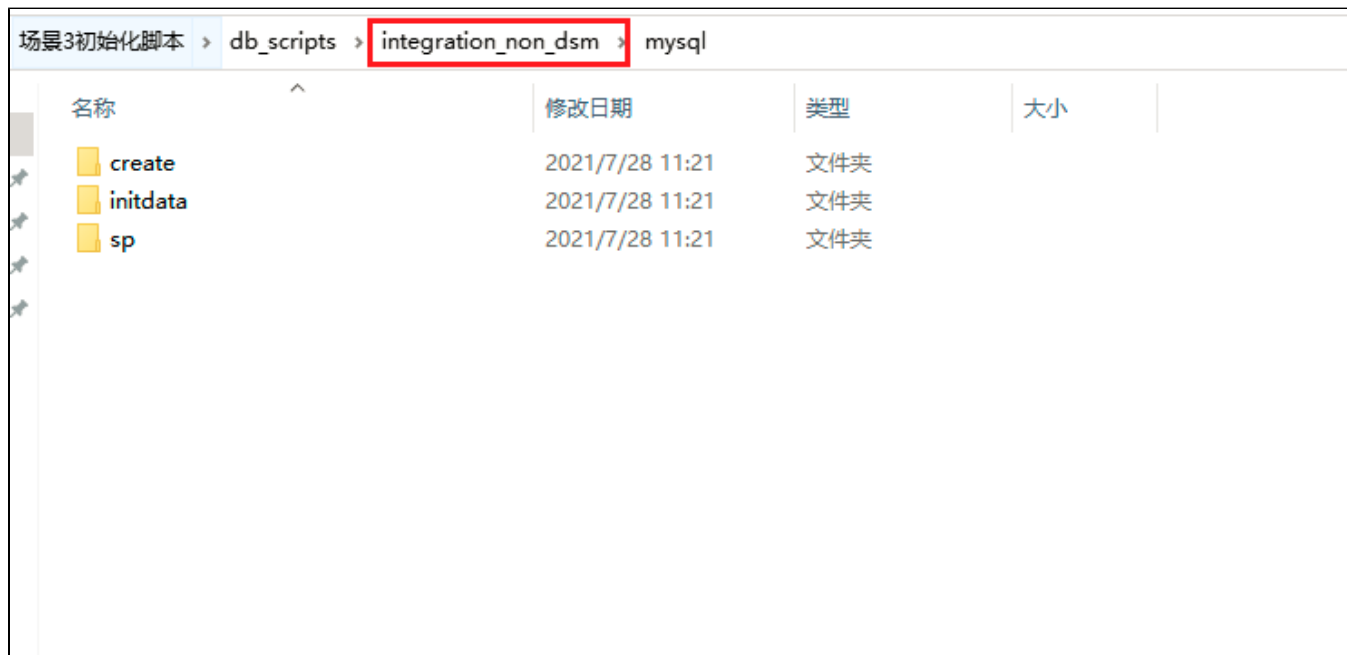


元数据脚本初始化

将元数据脚本 db_scripts.zip解压后，依次执行db_scripts\integration_non_dsm\mysql目录下所有子目录（create、initdata、sp）中的脚本。

说明

此示例中元数据作为集成应用和数据质量（基础应用）、数据资产目录做集成时，因当前集成环境中未部署数据标准，故使用integration_non_dsm目录下的脚本。



数据标准脚本初始化

将数据标准脚本 db_scripts.zip解压后，依次执行db_scripts\integration_with_metacube\mysql\initdata目录下所有脚本。

注意

此示例中数据标准作为集成应用和数据质量（基础应用）、数据资源目、元数据做集成时，因当前集成环境中已部署元数据，故使用 integration_with_metacube 目录下的脚本。

> 场景3初始化脚本 > 标准 > db_scripts > integration_with_metacube > mysql >				
	名称	修改日期	类型	大小
	initdata	2021/7/28 10:31	文件夹	

2.2 元数据管理平台安装部署

前置条件

- 已配置Redis服务。
- 已下载元数据管理平台安装介质。

软件包名称	软件包描述
Primeton_MetaCube_7.5LA_Server.tar	元数据管理平台DGS安装包
Primeton_MetaCube_7.5LA_Extractor.tar	元数据采集器安装包
Primeton_MetaCube_7.5LA_UI.tar	元数据前端安装包
db_scripts.zip	数据库初始化脚本

2.2.1 后端DGS服务部署

安装与配置

1. 创建“DGS”目录。
2. 解压安装包到 “DGS” 目录中。
 - Linux

```
#tar -xvf Primeton_MetaCube_7.5LA_Server.tar -C ./DGS
```

- Windows

用解压工具将 “Primeton_MetaCube_7.5LA_Server.tar” 解压至 “DGS” 目录。

3. 将数据库驱动包和待采集的数据库驱动包拷贝到“DGS\lib”目录中。
4. 在“DGS\conf\application.yml”文件，修改数据库配置参考如下。



需要修改的参数配置如下，其他项可保持默认值：

参数	说明
port: 8080	
context-path: /	修改为 /
redis: host: 192.168.30.144 password: primeton000000 port: 6379	redisip,
datasource: url: jdbc:mysql://localhost:3306/integration75?useUnicode=true&characterEncoding=UTF-8&useSSL=false&serverTimezone=Asia/Shanghai driverClassName: com.mysql.jdbc.Driver username: root password: root	数据库URL

coframe: cache: enabled: true type: memory key-namespace: integration167	typeredis key- namespace: key
--	--

启动

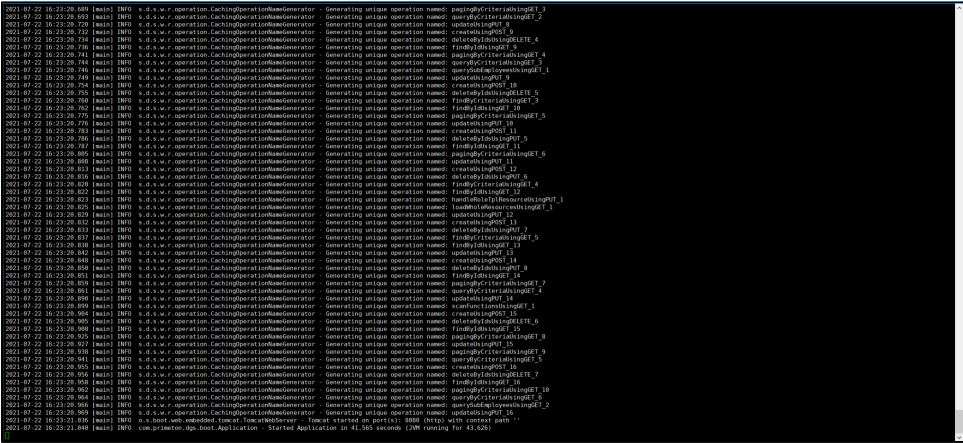
1. 在“DGS”目录下，执行如下命令，启动DGS。
- Linux

```
# ./startServer.sh
```

- Windows

```
> startServer.cmd
```

2. 启动完成，如下图所示。



说明

启动中后台若出现如下错误，需要在数据库连接url中ssl=true或者false即可，如下所示。

url=jdbc:mysql://127.0.0.1:3306/framework?useSSL=false

Fri Jul 30 18:48:58 CST 2021 WARN: Establishing SSL connection without server's identity verification is not recommended. According to MySQL 5.5.45+, 5.6.26+ and 5.7.6+ requirements SSL connection must be established by default if explicit option isn't set. For compliance with existing applications not using SSL, the verifyServerCertificate property is set to 'false'. You need either to explicitly disable SSL by setting useSSL=false, or set useSSL=true and provide truststore for server certificate verification.

卸载

1. 在“DGS”目录下，执行如下命令，停止元数据管理系统。
- Linux

```
# ./stopServer.sh
```

- Windows

```
> stopServer.cmd
```

2. 停止服务后，直接删除“DGS”目录来卸载。

2.2.2 元数据采集器部署

安装与配置

1. 创建“Extractor”目录。
2. 解压安装包到“Extractor”目录中。
 - Linux

```
# tar -xvf Primeton_MetaCube_7.5LA_Extractor.tar -C Extractor
```

- Windows

用解压工具将“Primeton_MetaCube_7.5LA_Extractor.tar”解压至“Extractor”目录

3. 将数据库驱动包和待采集的数据库驱动包拷贝到“Extractor\metacube_extractor\libs\3rd”目录中。
4. 在“Extractor\metacube_extractor\conf\extractor.properties”文件，修改采集器配置。

```
#

# IP
host=localhost      #IP

#
port=9999
.....
receivedFolderPath=      # Extractor\metacube_extractor\recFolder
```

启动

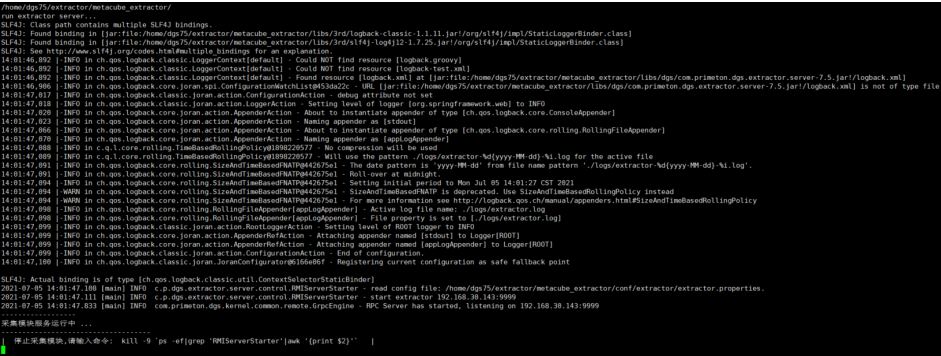
1. 在“Extractor”目录下，执行如下命令，启动采集器。
 - Linux

```
# nohup ./startExtractor.sh &
```

- Windows

```
> startExtractor.bat
```

启动完成后，如下图所示。



```
Primeton_MetaCube_7.5LA_Extractor
run extractor server...
SLF4J: Class path contains multiple SLF4J bindings.
SLF4J: Found binding in [jar:file:/home/dps/extractor/metacube_extractor/libs/3rd/logback-classic-1.1.11.jar!/org/slf4j/impl/StaticLoggerBinder.class]
SLF4J: Found binding in [jar:file:/home/dps/extractor/metacube_extractor/libs/3rd/slf4j-log4j12-1.7.25.jar!/org/slf4j/impl/StaticLoggerBinder.class]
SLF4J: See http://www.slf4j.org/codes.html#multiple_bindings for an explanation.
14:01:46.892 [INFO] in ch.qos.logback.classic.LoggerContext[default]: Could NOT find resource [logback-test.xml]
14:01:46.892 [INFO] in ch.qos.logback.classic.LoggerContext[default]: Found resource [logback.xml] at [jar:file:/home/dps/extractor/metacube_extractor/libs/dps.com.primeton.dps.extractor.server-7.5.jar!/logback.xml]
14:01:46.906 [INFO] in ch.qos.logback.core.joran.action.ConfigurationAction: About to instantiate appender of type [ch.qos.logback.core.ConsoleAppender]
14:01:47.017 [INFO] in ch.qos.logback.classic.joran.action.ConfigurationAction: debug attribute not set
14:01:47.018 [INFO] in ch.qos.logback.classic.joran.action.LoggerAction: Setting level of logger [org.springframework.web] to INFO
14:01:47.020 [INFO] in ch.qos.logback.core.joran.action.AppenderAction: Naming appender as [stdout]
14:01:47.064 [INFO] in ch.qos.logback.core.joran.action.AppenderAction: About to instantiate appender of type [ch.qos.logback.core.rolling.RollingFileAppender]
14:01:47.070 [INFO] in ch.qos.logback.core.joran.action.AppenderAction: Naming appender as [applogappender]
14:01:47.088 [INFO] in ch.qos.logback.core.rolling.RollingFileAppender: No compression will be used
14:01:47.089 [INFO] in ch.qos.logback.core.rolling.RollingFileAppender: Will use the pattern ./log/extractor-%d{yyyy-MM-dd}-%i.log for the active file
14:01:47.091 [INFO] in ch.qos.logback.core.rolling.RollingFileAppender: The data pattern is yyyy-MM-dd from file name pattern ./log/extractor-%d{yyyy-MM-dd}-%i.log.
14:01:47.091 [INFO] in ch.qos.logback.core.rolling.RollingFileAppender: Roll-over at midnight.
14:01:47.094 [INFO] in ch.qos.logback.core.rolling.RollingFileAppender: Setting initial period to Mon Jul 05 14:01:27 CST 2021.
14:01:47.094 [WARN] in ch.qos.logback.core.rolling.RollingFileAppender: SizeAndTimeBasedRollingPolicy is deprecated. Use SizeAndTimeBasedRollingPolicy instead
14:01:47.094 [WARN] in ch.qos.logback.core.rolling.RollingFileAppender: For more information see http://logback.qos.ch/manual/appenders.html#sizeandtimebasedrollingpolicy
14:01:47.099 [INFO] in ch.qos.logback.core.rolling.RollingFileAppender: Active log file name: ./log/extractor.log
14:01:47.099 [INFO] in ch.qos.logback.classic.joran.action.RootLoggerAction: Setting level of ROOT logger to INFO
14:01:47.099 [INFO] in ch.qos.logback.core.joran.action.AppenderRefAction: Attaching appender named [stdout] to logger[ROOT]
14:01:47.099 [INFO] in ch.qos.logback.core.joran.action.AppenderRefAction: Attaching appender named [applogappender] to logger[ROOT]
14:01:47.099 [INFO] in ch.qos.logback.classic.joran.action.ConfigurationAction: End of configuration.
14:01:47.100 [INFO] in ch.qos.logback.classic.joran.action.ConfigurationAction: Registering current configuration as safe fallback point

SLF4J: Actual binding is of type [ch.qos.logback.classic.util.ContextSelectorStaticBinder]
2021-07-05 14:01:47.111 [main] INFO c.p.dps.extractor.server.control.HMSEServerStarter - read config file: /home/dps/extractor/metacube_extractor/conf/extractor/extractor.properties.
2021-07-05 14:01:47.133 [main] INFO c.p.dps.extractor.server.control.HMSEServerStarter - start extractor 192.168.30.143:9999
2021-07-05 14:01:47.133 [main] INFO c.p.primeton.dps.kernel.common.remote.Engine - DPS Server has started, listening on 192.168.30.143:9999

元数据抽取器启动中 ...
停止元数据抽取器，请输入命令：kill -9 ps -ef|grep 'HMSEServerStarter'|awk '{print $2}' |
```

卸载

直接删除“Extractor”目录来卸载。

2.3 数据资产目录安装部署

前置条件

- 已配置Redis服务。
- 已下载数据资产目录安装介质。

软件包名称	软件包描述
Primeton_DataCatalogCube_7.0LA_OnlineConfig.tar	在线页面配置服务安装包
Primeton_DataCatalogCube_7.0LA_DataRelease.tar	数据服务发布安装包
Primeton_DataCatalogCube_7.0LA_Server.tar	数据资产目录DAMP安装包
Primeton_DataCatalogCube_7.0LA_UI.tar	数据资产目录前端安装包

安装须知

数据资产目录安装介质中，在线页面配置服务安装包（Primeton_DataCatalogCube_7.0LA_OnlineConfig.tar）、数据资产目录安装包（Primeton_DataCatalogCube_7.0LA_Server.tar）、数据资产目录前端安装包（Primeton_DataCatalogCube_7.0LA_UI.tar）需要部署在同一台服务器。

2.3.1 在线页面配置部署

安装

1. 创建damp_onlineConfig目录，在damp_onlineConfig目录下解压安装包。

- Linux

```
#tar -xvf Primeton_DataCatalogCube_7.0LA_OnlineConfig.tar -C ./damp_onlineConfig
```

- Windows

```
用解压工具将 “Primeton_DataCatalogCube_7.0LA_OnlineConfig.tar” 解压至 “damp_onlineConfig” 目录
```

2. 在damp_onlineConfig/apache-tomcat-8.5.29/webapps/ROOT/WEB-INF/classes目录下修改 jdbc.properties文件。

```
jdbc.database=MySQL
jdbc.driverClassName=com.mysql.jdbc.Driver
jdbc.url=jdbc:mysql://localhost:3306/integration75?characterEncoding=utf-8&useOldAliasMetadataBehavior=true&autoReconnect=true #
jdbc.username=root #
jdbc.password=root #
jndi.enabled=false
jndi.name=java:comp/env/jdbc/Datasource_MDMS
```

主要配置说明如下：其他参数使用默认配置即可。

参数	说明
jdbc.url=jdbc:mysql://localhost:3306/integration75?characterEncoding=utf-8&... jdbc.username=root jdbc.password=123456	

启动

1. 在damp_onlineConfig/apache-tomcat-8.5.29/bin目录下执行如下命令，启动在线页面服务配置。

- Linux

```
# ./startup.sh
```

- Windows

```
> startup.bat
```

访问验证



2.3.2 数据服务部署

安装

1. 在damp_DataRelease目录下解压安装包。

- Linux

```
#tar -xvf Primeton_DataCatalogCube_7.0LA_DataRelease.tar -C ./damp_DataRelease
```

- Windows

用解压工具将 “Primeton_DataCatalogCube_7.0LA_DataRelease.tar” 解压至 “damp_DataRelease” 目录

2. 将数据库驱动包拷贝到 “damp_DataRelease\lib” 目录中。

3. 修改damp_DataRelease/conf目录下application.yml文件。

```
server:
  port: 19189          #
  servlet:
    context-path: /datapub

mybatis:
  #
  type-aliases-package: com.primeton.damp.datapub.entity
  # mapperBase
  mapper-locations: classpath:mapper/mysql/*Mapper.xml
  configuration:
    call-setters-on-nulls: true

common:
  jdbc:
    driverClass: com.mysql.jdbc.Driver
    username: "root"          ###-
    password: "123456"        ###-
  #
  base_dir: ${app.home:.}
  # DAMP
  damp_dir: ${app.home:.}    ###-{Primeton_DataCatalogCube_7.0LA_Server}

logging:
  #
  level.com.primeton.damp: debug
  level.com.primeton.dqms: debug

spring:
  application:
    name: datapub
  jackson:
    date-format: yyyy-MM-dd HH:mm:ss
    time-zone: GMT+8
  datasource:
    type: com.alibaba.druid.pool.DruidDataSource
    url: jdbc:mysql://192.168.30.142:3306/coframe6?useUnicode=true&characterEncoding=UTF-8&useSSL=false #
    driverClassName: ${common.jdbc.driverClass}
    username: ${common.jdbc.username}
    password: ${common.jdbc.password}
    druid:
      initialSize: 5
      maxActive: 50
      maxWait: 60000
      minIdle: 1
      timeBetweenEvictionRunsMillis: 60000
      minEvictableIdleTimeMillis: 300000
      validationQuery: SELECT count(*) from EOS_UNIQUE_TABLE
      testWhileIdle: true
      testOnBorrow: false
      testOnReturn: false
```

```
poolPreparedStatements: true
maxPoolPreparedStatementPerConnectionSize: 20
filter:
  slf4j:
    enabled: true
  stat:
    enabled: true

# \u5206\u9875\u63D2\u4EF6
# https://github.com/pagehelper/Mybatis-PageHelper/blob/master/wikis/zh/HowToUse.md
pagehelper:
  page-size-zero: true
  helperDialect: mysql
  # helperDialect: oracle
  reasonable: true
  supportMethodsArguments: true

datapub:
  async:
    poll:
      corePoolSize: 5
      maxPoolSize: 50
      keepAliveSeconds: 60
      queueCapacity: 10000
      threadNamePrefix: DP-AsyncThread-
  datasource:
    poll:
      initialSize: 3
      minIdle: 3
      maxActive: 8
      maxWait: 60000

logwriter:
  enable: true
  exclude:
    - /favicon.ico
    - /refresh
  buffer:
    threshold: 10
    max: 20
    flush: 1000
  file:
    path: logs/datapub
    name: dsp.log
    roll: 6000
    size: 2048000
  record:
    enable: true
    columnseparator: 18
    lineseparator: 17
  writebody:
    iswritebody: true
    iswritemultipart: true
```

主要配置说明如下：其他参数使用默认配置即可。

参数	说明
server: port: 19189	
common: jdbc: driverClass: com.mysql.jdbc.Driver username: "root" password: "123456"	

```
datasource:
  type: com.alibaba.druid.pool.DruidDataSource
  url: jdbc:mysql://192.168.30.142:3306/coframe6?useUnicode=true&characterEncoding=UTF-8&useSSL=false
```

启动

1. 在damp_DataRelease目录下，执行如下命令启动，日志在logs目录下查看。

- Linux

```
# ./startDatapub.sh
```

- Windows

```
> startDatapub.cmd
```

```

.mapper.ServiceTimeoutMapper' mapperInterface. Bean already defined with the same name!
2021-07-06 15:48:25 [main] - [ WARN ] [ org.mybatis.spring.mapper.ClassPathMapperScanner : 44 ] - Skipping MapperFactoryBean with name 'serviceValidateInfoMapper' and 'com.primeton.damp.da
tapub.mapper.ServiceValidateInfoMapper' mapperInterface. Bean already defined with the same name!
2021-07-06 15:48:25 [main] - [ WARN ] [ org.mybatis.spring.mapper.ClassPathMapperScanner : 44 ] - Skipping MapperFactoryBean with name 'whiteIPListMapper' and 'com.primeton.damp.datapub.ma
pper.WhiteIPListMapper' mapperInterface. Bean already defined with the same name!
2021-07-06 15:48:25 [main] - [ WARN ] [ org.mybatis.spring.mapper.ClassPathMapperScanner : 44 ] - No MyBatis mapper was found in '[com.primeton.damp.datapub]' package. Please check your co
nfiguration.
2021-07-06 15:48:25 [main] - [ INFO ] [ c.u.j.c.EnableEncryptablePropertiesBeanFactoryPostProcessor : 50 ] - Post-processing PropertySource instances
2021-07-06 15:48:25 [main] - [ INFO ] [ c.u.j.EncryptablePropertySourceConverter : 38 ] - Converting PropertySource configurationProperties [org.springframework.boot.context.properties.sou
rce.ConfigurationPropertySource$PropertySource] to AOP Proxy
2021-07-06 15:48:25 [main] - [ INFO ] [ c.u.j.EncryptablePropertySourceConverter : 38 ] - Converting PropertySource servletConfigInitParams [org.springframework.core.env.PropertySource$Stu
bPropertySource] to EncryptablePropertySourceWrapper
2021-07-06 15:48:25 [main] - [ INFO ] [ c.u.j.EncryptablePropertySourceConverter : 38 ] - Converting PropertySource servletContextInitParams [org.springframework.core.env.PropertySource$St
ubPropertySource] to EncryptablePropertySourceWrapper
2021-07-06 15:48:25 [main] - [ INFO ] [ c.u.j.EncryptablePropertySourceConverter : 38 ] - Converting PropertySource systemProperties [org.springframework.core.env.PropertiesPropertySource]
to EncryptableMapPropertySourceWrapper
2021-07-06 15:48:25 [main] - [ INFO ] [ c.u.j.EncryptablePropertySourceConverter : 38 ] - Converting PropertySource systemEnvironment [org.springframework.boot.env.SystemEnvironmentPropert
ySource$EnvironmentPostProcessor$OriginAwareSystemEnvironmentPropertySource] to EncryptableMapPropertySourceWrapper
2021-07-06 15:48:25 [main] - [ INFO ] [ c.u.j.EncryptablePropertySourceConverter : 38 ] - Converting PropertySource random [org.springframework.boot.env.RandomValuePropertySource] to Encry
ptablePropertySourceWrapper
2021-07-06 15:48:25 [main] - [ INFO ] [ c.u.j.EncryptablePropertySourceConverter : 38 ] - Converting PropertySource applicationConfig: [classpath:/application.yml] [org.springframework.boo
t.env.OriginTrackedMapPropertySource] to EncryptableMapPropertySourceWrapper
2021-07-06 15:48:25 [main] - [ INFO ] [ c.u.j.EncryptablePropertySourceConverter : 38 ] - Converting PropertySource class path resource [application.yml] [org.springframework.core.io.supp
rt.ResourcePropertySource] to EncryptableMapPropertySourceWrapper
2021-07-06 15:48:25 [main] - [ INFO ] [ c.u.j.filter.DefaultLazyPropertyFilter : 31 ] - Property Filter custom Bean not found with name 'encryptablePropertyFilter'. Initializing Default Pr
operty Filter
2021-07-06 15:48:25 [main] - [ INFO ] [ o.s.c.s.PostProcessorRegistrationDelegatesBeanPostProcessorChecker : 330 ] - Bean 'org.springframework.transaction.annotation.ProxyTransactionManage
mentConfiguration' of type [org.springframework.transaction.annotation.ProxyTransactionManagementConfiguration$EnhancerBySpringCdiIb$5id11382b] is not eligible for getting processed by all B
eanPostProcessors (for example: not eligible for auto-proxying)
2021-07-06 15:48:25 [main] - [ INFO ] [ c.u.j.resolver.DefaultLazyPropertyResolver : 31 ] - Property Resolver custom Bean not found with name 'encryptablePropertyResolver'. Initializing De
fault Property Resolver
2021-07-06 15:48:25 [main] - [ INFO ] [ c.u.j.detector.DefaultLazyPropertyDetector : 30 ] - Property Detector custom Bean not found with name 'encryptablePropertyDetector'. Initializing De
fault Property Detector
2021-07-06 15:48:25 [main] - [ INFO ] [ o.s.boot.web.embedded.tomcat.TomcatWebServer : 90 ] - Tomcat initialized with port(s): 19189 (http)
2021-07-06 15:48:25 [main] - [ INFO ] [ org.apache.coyote.http11.Http11NioProtocol : 173 ] - Initializing ProtocolHandler ["http-nio-19189"]
2021-07-06 15:48:25 [main] - [ INFO ] [ org.apache.catalina.core.StandardService : 173 ] - Starting service [Tomcat]
```

卸载

1. 在“damp_DataRelease”目录下，执行如下命令，停止数据服务发布。

- Linux

```
# ./stopDatapub.sh
```

- Windows

```
> stopDatapub.sh
```

2. 停止服务后，直接删除“damp_DataRelease”目录来卸载。

2.3.3 后端Server部署

配置服务

1. 创建 “damp_server” 目录, 解压安装包到 “damp_server” 目录中。
 - Linux

```
#tar -xvf Primeton_DataCatalogCube_7.0LA_Server.tar -C ./damp_server
```

- Windows

用解压工具将 “Primeton_DataCatalogCube_7.0LA_Server.tar” 解压至 “damp_server” 目录

2. 将数据库驱动包拷贝到 “damp_server\lib” 目录中。
3. 在damp_server/conf目录下修改application-prod.yml文件。

说明



配置文件中需要修改的地方在右侧使用 “###-” 备注说明。其他默认可以不做修改。

```
server:
  port: 8089                                     ###-
  servlet:
    context-path: /

common:
  jdbc:
    url: "jdbc:mysql://localhost:3306/db_name?useUnicode=true&characterEncoding=utf-8"  ###-
    driverClass: com.mysql.jdbc.Driver
    host: ""
    username: "root"                             ###-
    password: "****"                             ###-
    alias: "db_name"                             ###-

db_name
  base_dir: ${app.home:.}

spring:
  redis:
    host: 192.168.30.144                         ###-redisip
    port: 6379                                    ###-redis
    password: primeton000000                      ###-redis

freemarker:
  cache: false  #

#
datasource:
  # # SQL allowMultiQueries=true
  coframe:
    # jdbc-urlurl
    #url: jdbc:oracle:thin:@192.168.5.101:1521:orcl
    url: ${common.jdbc.url}
    # root
    username: ${common.jdbc.username}
    # root
    password: ${common.jdbc.password}
    #driver-class-name: oracle.jdbc.OracleDriver
    driver-class-name: ${common.jdbc.driverClass}
    alias: ${common.jdbc.alias}
    #
    initial-size: 10
    #
    min-idle: 10
    #
    max-active: 20
  damp:
    # jdbc-urlurl
    #url: jdbc:oracle:thin:@192.168.5.101:1521:orcl
    url: ${common.jdbc.url}
    # root
    username: ${common.jdbc.username}
    # root
    password: ${common.jdbc.password}
```

```

#
#driver-class-name: oracle.jdbc.OracleDriver
driver-class-name: ${common.jdbc.driverClass}
alias: ${common.jdbc.alias}
#
initial-size: 10
#
min-idle: 10
#
max-active: 20
dqms:
#dqms
#
url: ${common.jdbc.url}
# root
username: ${common.jdbc.username}
# root
password: ${common.jdbc.password}
#
driver-class-name: ${common.jdbc.driverClass}
alias: ${common.jdbc.alias}
#
initial-size: 10
#
min-idle: 10
#
max-active: 20
metacube:
enabled: true
# jdbc-urlurl
#url: jdbc:oracle:thin:@127.0.0.1:1521:orcl
url: ${common.jdbc.
url}
# root
username: ${common.jdbc.username}
###-
###-
#
password:
" * * * "
###-
#
driver-class-name: ${common.jdbc.driverClass}
#driver-class-name: oracle.jdbc.OracleDriver
#schema
alias: ${common.jdbc.alias}
###-
#
initial-size: 2
#
min-idle: 2
#
max-active: 5

druid:
#
max-wait: 5000
#
time-between-connect-error-millis: 10000

jpa:
open-in-view: false
cache:
#
type: memory
redis:
# key
key-prefix: dampDev
#
time-to-live: 36000s

servlet:
# http://ip:port/files/xxx.txt

```

```

staticAccessPath: /files/**
#uploadPath: /home/damp/uploadFiles/
uploadPath: ${common.base_dir}/upload/principal
multipart:
  # ( true)
  enabled: true
  # realfile.max-size 5000M 1M springboot
  max-file-size: 5000MB
  # 5000M10M springboot
  max-request-size: 5000MB
# solr
solr:
  solrUser: solr
  solrPassWord: solr!123
  serverUrl: http://192.168.30.142:17000/solr      ###-solrsolr
  collectionName: dgp
  #
  commitLimit: 5000
  #
  copyLimit: 500
  #
  threadNum: 20

logging:
  #
  level.com.primeton.damp: INFO
  level.com.primeton.dqms: INFO

# Json web token
jwt:
  #
  expire-time: 600m
  # claim key
  claim-key-auth: auth
  # key
  header: Authorization
  # token
  token-type: Bearer

hadoop:
  # hadoop HDFScluster, local
  hdfs_type: local
  tenent_name: ""
  core-site: "${common.base_dir}/conf/core-site.xml"
  hdfs-site: "${common.base_dir}/conf/hdfs-site.xml"
  user_keytab: "${common.base_dir}/conf/user.keytab"
  krb5_file: "${common.base_dir}/conf/krb5.conf"
  # HDFS
  fileDownloadUrl: "http://192.168.30.140:50070"

cache:
  type: memory

datashard:
  # localftpsftphdfs
  type: local
  base_path: ${common.base_dir}/upload
  #hadoop
  delSrc: true
  #
  fileSuffix: .bcp

ftp:
  host: localhost
  port: 21
  userName: admin
  password: primeton0000000

sftp:

```

```
# ip
host: localhost
#
port: 22
#
userName: root
#
password: primeton

file:
  dqms:
    #
    otherErrorDataFilePath: ${common.base_dir}/dataCheck/otherErrorData
    #
    dataSummaryFilePath: ${common.base_dir}/dataCheck/dataSummary
    #
    errorDetailFilePath: ${common.base_dir}/dataCheck/errorDetail
  damp:
    #
    #dataTemplateFilePath: /home/resourceTemplate/dataResourceImport.xlsx
    dataTemplateFilePath: ${common.base_dir}/resourceTemplate/resourceTemplate.xlsx
    # 5M
    templateFileLocation: ${common.base_dir}/fileTemplate/dataResourceImport.xlsx
    #
    VerificationTemplatePath: ${common.base_dir}/resourceTemplate/VerificationTemplate.xlsx
    ciclassTemplateFilePath: ${common.base_dir}/resourceTemplate/classifDataImport.xlsx

#
metacube:
  version: 75          ###-
  address: http://192.168.30.143:18780/dgs    ###- dgs

# /
datapub:
  address: http://192.168.30.142:19189/datapub    ###-

datacheck:
  DBkerberos:
    #po_filefilepath
    path: ${common.base_dir}/kerberos/

#
page-field:
  url: http://192.168.30.142:28180    ###-
```

主要配置说明如下：其他参数使用默认配置即可。

参数	说明
server: port:	
common: jdbc: url: jdbc:mysql://localhost:3306/coframe5?useUnicode=true&characterEncoding=utf-8 driverClass: com.mysql.jdbc.Driver host: "" username: "root" password: "" alias: "db_name" #db_name	URI
redis: host: localhost port: 6379 password: primeton000000	redisipredis

说明



若在server启动后修改上述数据库参数值，则需要重新启动damp_server。

6. 若application-prod.yml文件中配置的数据库用户无mysql库下help_topic表的查询权限时，需要赋查询权限。
GRANT SELECT ON mysql.help_topic TO 'user'@'localhost';

启动

1. 在“damp_server”目录下，执行如下命令启动服务，日志在logs目录下查看。

- Linux

```
# ./startServer.sh
```

- Windows

```
> startServer.cmd
```

完成启动后，如下图所示。

```
2021-07-06 14:49:37 [main] - [INFO] [s.d.s.w.r.operation.CachingOperationNameGenerator : 40] - Generating unique operation named: listUsingGET_26
2021-07-06 14:49:37 [main] - [INFO] [s.d.s.w.r.operation.CachingOperationNameGenerator : 40] - Generating unique operation named: saveUsingPOST_1
2021-07-06 14:49:37 [main] - [INFO] [s.d.s.w.r.operation.CachingOperationNameGenerator : 40] - Generating unique operation named: updateUsingPUT_38
2021-07-06 14:49:37 [main] - [INFO] [s.d.s.w.r.operation.CachingOperationNameGenerator : 40] - Generating unique operation named: queryMappingTotalUsingGET_1
2021-07-06 14:49:37 [main] - [INFO] [s.d.s.w.r.operation.CachingOperationNameGenerator : 40] - Generating unique operation named: addUsingPOST_24
2021-07-06 14:49:37 [main] - [INFO] [s.d.s.w.r.operation.CachingOperationNameGenerator : 40] - Generating unique operation named: deleteUsingDELETE_27
2021-07-06 14:49:37 [main] - [INFO] [s.d.s.w.r.operation.CachingOperationNameGenerator : 40] - Generating unique operation named: detailUsingGET_23
2021-07-06 14:49:37 [main] - [INFO] [s.d.s.w.r.operation.CachingOperationNameGenerator : 40] - Generating unique operation named: listUsingGET_27
2021-07-06 14:49:37 [main] - [INFO] [s.d.s.w.r.operation.CachingOperationNameGenerator : 40] - Generating unique operation named: updateUsingPUT_39
2021-07-06 14:49:37 [main] - [INFO] [s.d.s.w.r.operation.CachingOperationNameGenerator : 40] - Generating unique operation named: listUsingGET_28
2021-07-06 14:49:37 [main] - [INFO] [s.d.s.w.r.operation.CachingOperationNameGenerator : 40] - Generating unique operation named: saveNoticeUsingPOST_1
2021-07-06 14:49:37 [main] - [INFO] [s.d.s.w.r.operation.CachingOperationNameGenerator : 40] - Generating unique operation named: updateNoticeUsingPUT_1
2021-07-06 14:49:37 [main] - [INFO] [s.d.s.w.r.operation.CachingOperationNameGenerator : 40] - Generating unique operation named: createUsingPOST_14
2021-07-06 14:49:37 [main] - [INFO] [s.d.s.w.r.operation.CachingOperationNameGenerator : 40] - Generating unique operation named: deleteByIdsUsingPUT_8
2021-07-06 14:49:37 [main] - [INFO] [s.d.s.w.r.operation.CachingOperationNameGenerator : 40] - Generating unique operation named: findByIdUsingGET_14
2021-07-06 14:49:37 [main] - [INFO] [s.d.s.w.r.operation.CachingOperationNameGenerator : 40] - Generating unique operation named: pagingByCriteriaUsingGET_7
2021-07-06 14:49:37 [main] - [INFO] [s.d.s.w.r.operation.CachingOperationNameGenerator : 40] - Generating unique operation named: queryByCriteriaUsingGET_4
2021-07-06 14:49:37 [main] - [INFO] [s.d.s.w.r.operation.CachingOperationNameGenerator : 40] - Generating unique operation named: updateUsingPUT_40
2021-07-06 14:49:37 [main] - [INFO] [s.d.s.w.r.operation.CachingOperationNameGenerator : 40] - Generating unique operation named: scanFunctionsUsingGET_1
2021-07-06 14:49:37 [main] - [INFO] [s.d.s.w.r.operation.CachingOperationNameGenerator : 40] - Generating unique operation named: createUsingPOST_15
2021-07-06 14:49:37 [main] - [INFO] [s.d.s.w.r.operation.CachingOperationNameGenerator : 40] - Generating unique operation named: deleteByIdsUsingDELETE_6
2021-07-06 14:49:37 [main] - [INFO] [s.d.s.w.r.operation.CachingOperationNameGenerator : 40] - Generating unique operation named: findByIdUsingGET_15
2021-07-06 14:49:37 [main] - [INFO] [s.d.s.w.r.operation.CachingOperationNameGenerator : 40] - Generating unique operation named: pagingByCriteriaUsingGET_8
2021-07-06 14:49:37 [main] - [INFO] [s.d.s.w.r.operation.CachingOperationNameGenerator : 40] - Generating unique operation named: updateUsingPUT_41
2021-07-06 14:49:37 [main] - [INFO] [s.d.s.w.r.operation.CachingOperationNameGenerator : 40] - Generating unique operation named: pagingByCriteriaUsingGET_9
2021-07-06 14:49:37 [main] - [INFO] [s.d.s.w.r.operation.CachingOperationNameGenerator : 40] - Generating unique operation named: queryByCriteriaUsingGET_5
2021-07-06 14:49:37 [main] - [INFO] [s.d.s.w.r.operation.CachingOperationNameGenerator : 40] - Generating unique operation named: createUsingPOST_16
2021-07-06 14:49:37 [main] - [INFO] [s.d.s.w.r.operation.CachingOperationNameGenerator : 40] - Generating unique operation named: deleteByIdsUsingDELETE_7
2021-07-06 14:49:37 [main] - [INFO] [s.d.s.w.r.operation.CachingOperationNameGenerator : 40] - Generating unique operation named: findByIdUsingGET_16
2021-07-06 14:49:37 [main] - [INFO] [s.d.s.w.r.operation.CachingOperationNameGenerator : 40] - Generating unique operation named: pagingByCriteriaUsingGET_10
2021-07-06 14:49:37 [main] - [INFO] [s.d.s.w.r.operation.CachingOperationNameGenerator : 40] - Generating unique operation named: queryByCriteriaUsingGET_6
2021-07-06 14:49:37 [main] - [INFO] [s.d.s.w.r.operation.CachingOperationNameGenerator : 40] - Generating unique operation named: querySubEmployeesUsingGET_2
2021-07-06 14:49:37 [main] - [INFO] [s.d.s.w.r.operation.CachingOperationNameGenerator : 40] - Generating unique operation named: updateUsingPUT_42
2021-07-06 14:49:37 [main] - [INFO] [o.s.boot.web.embedded.tomcat.TomcatWebServer : 202] - Tomcat started on port(s): 17777 (http) with context path '/damp'
2021-07-06 14:49:37 [main] - [INFO] [com.primeton.damp.Application : 59] - Started Application in 19.585 seconds (JVM running for 20.148)
```

说明



启动中后台若出现如下错误，需要在数据库连接url中加入useSSL=true或者false即可，如下所示。

url=jdbc:mysql://127.0.0.1:3306/framework?useSSL=false

```
Fri Jul 30 18:40:50 CST 2021 WARN: Establishing SSL connection without server's identity verification is not recommended. According to MySQL 5.5.45+, 5.6.26+ and 5.7.6+ requirements SSL connection must be established by default if explicit option isn't set. For compliance with existing applications not using SSL the verifyServerCertificate property is set to 'false'. You need either to explicitly disable SSL by setting useSSL=false, or set useSSL=true and provide truststore for server certificate verification.
```

卸载

1. 在“damp_server”目录下，执行如下命令，停止数据资产目录。

- Linux

```
# ./stopServer.sh
```

- Windows

```
> stopServer.cmd
```

2. 停止服务后，直接删除“damp_server”目录来卸载。

2.4 数据标准管理平台安装部署

前置条件

- 已配置Redis服务。
- 已下载数据标准管理平台安装介质。

软件包名称	软件包描述
Primeton_DataStandardCube_7.0LA_Server.tar	数据标准管理平台DSM安装包
Primeton_DataStandardCube_7.0LA_Extractor.tar	数据标准采集器安装包
Primeton_DataStandardCube_7.0LA_UI.tar	数据标准前端安装包
db_scripts.zip	数据库初始化脚本

2.4.1 后端DSM服务部署

安装与配置

1. 创建“DSM”目录。
2. 解压安装包到“DSM”目录中。
 - Linux

```
#tar -xvf Primeton_DataStandardCube_7.0LA_Server.tar -C ./DSM
```

- Windows

用解压工具将“Primeton_DataStandardCube_7.0LA_Server.tar”解压至“DSM”目录。

3. 将数据库驱动包和待采集的数据库驱动包拷贝到“DSM\lib”目录中。
4. 在“DSM\conf\application.yml”文件，修改数据库配置，参考下面的配置。

```
server:
  port: 8088 #
  tomcat:
    max-threads: 200
  servlet:
    context-path: / ### /

spring:
  application:
    name: DSM
  redis:
    host: 192.168.30.144 #redis
    password: primeton000000 #redis
    port: 6379 #redisip
  jackson:
    date-format: yyyy-MM-dd HH:mm:ss
    time-zone: GMT+8
  datasource:
    url: jdbc:mysql://localhost:3306/integration75?useUnicode=true&characterEncoding=UTF-8&useSSL=false&serverTimezone=UTC #URL
    driverClassName: com.mysql.jdbc.Driver
    username: root #
    password: root #
    initialSize: 5
    maxActive: 50
    maxWait: 60000
    minIdle: 5
    timeBetweenEvictionRunsMillis: 60000
    minEvictableIdleTimeMillis: 300000
    validationQuery: SELECT 1
    testWhileIdle: true
    testOnBorrow: false
    testOnReturn: false
    poolPreparedStatements: true
    maxPoolPreparedStatementPerConnectionSize: 20
    filter:
      slf4j:
        enabled: true
      stat:
        enabled: true
  resources:
    static-locations: ./webapp
  jpa:
    open-in-view: false
  mvc:
    view:
      prefix: /
      suffix: .jsp
  freemarker:
    template-loader-path: [classpath:/template/notify]
```

```

eureka:
  client:
    enabled: false

eos:
  cache:
    RULES_IP_CACHE:
      mode: map
  admin:
    port: 6200

dgs:
  threadpool:
    corePoolSize: 5
    maxPoolSize: 50
    keepAliveSeconds: 60
    queueCapacity: 10000
    threadNamePrefix: DGS-ThreadPool-

pagehelper:
  auto-dialect: mysql
  row-bounds-with-count: true

swagger:
  group: dsm
  title: DSM
  description: DSM
  version: 7.5.0
  creator: primeton
  basepackage: com.primeton.standard

coframe:
  cache:
    enabled: true
    type: memory
    key-namespace: integraation167
    #
  auth: #
    token-expired: 7200 # token
    token-refresh-interval: 1200 # token
    relogin-keep-token: false # , token, false
    exclude-paths: /images/**,/fonts/**,/js/**, **/login, /druid/**, /rest/**, /console/**,*.js,*.do,*.css,*.png,*.gif,*.jpg,*.jpeg # url, .
  bean-enabled: # bean
    hibernate5-module: true # Hibernate5Module, true
    multipart-config-factory: true # MultipartConfigElement, true
    swagger-docket: false # coframe swagger Docket, true
    local-validator-factory-bean: true # LocalValidatorFactoryBean, true
    exception-handler-all: true # coframe, , true
  flag:
    employee-user-cascade-delete: false # ,
  filter: #
    token: # token
    enabled: false # sdktoken, true. coframe-coreJWTFilter, false
  tools:
    enabled: true # . , , , . true, false

```



需要修改的参数配置如下，其他项可保持默认值：

参数	说明
port: 8080	
context-path: /	修改为 /
redis: host: 192.168.30.144 password: primeton000000 port: 6379	redisip,
datasource: url: jdbc:mysql://localhost:3306/integration75?useUnicode=true&characterEncoding=UTF-8&useSSL=false&serverTimezone=Asia/Shanghai driverClassName: com.mysql.jdbc.Driver username: root password: root	数据库URL
coframe: cache: enabled: true type: memory key-namespace: integration167	typeredis key-namespace: key

启动

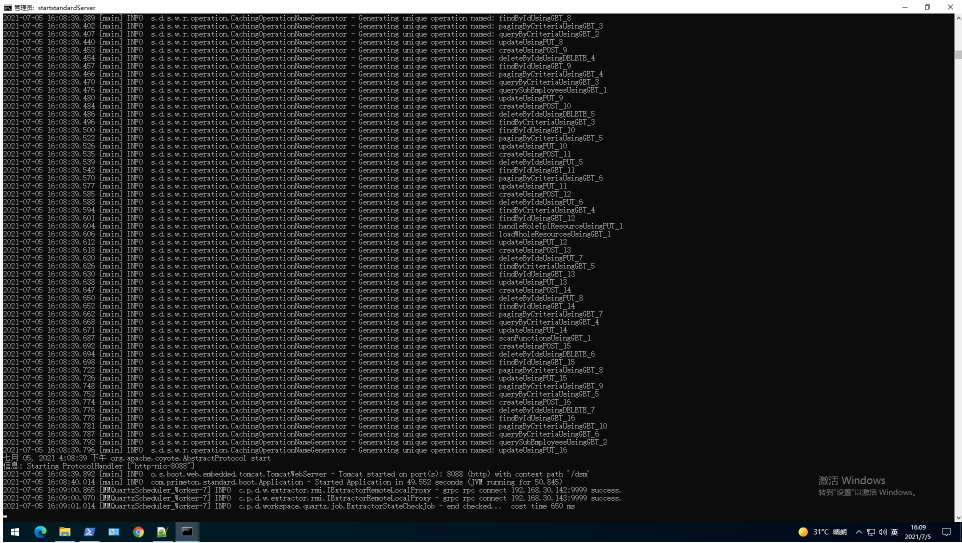
1. 在“DSM”目录下，执行如下命令，启动DSM。

- Linux

```
# ./startServer.sh
```

- Windows

```
> startServer.cmd
```



2. 启动完成，如下图所示。

卸载

1. 在“DSM”目录下，执行如下命令，停止数据标准管理平台。

- Linux

```
# ./stopServer.sh
```

- Windows

```
> stopServer.cmd
```

2. 停止服务后，直接删除“DSM”目录来卸载。

2.4.2 数据标准采集器部署

安装与配置

1. 创建“dsm_Extractor”目录。
2. 解压安装包到 “dsm_Extractor” 目录中。
 - Linux

```
# tar -xvf Primeton_DataStandardCube_7.0LA_Extractor.tar -C ./dsm_Extractor
```

- Windows

用解压工具将 “Primeton_DataStandardCube_7.0LA_Extractor.tar” 解压至 “dsm_Extractor” 目录。

3. 将数据库驱动包拷贝到 “dsm_Extractor\metacube_extractor\libs\3rd” 目录中。
4. 在 “dsm_Extractor\standard_extractor\conf\extractor\extractor.properties” 文件，修改采集器配置。

```
#

# IP
host=localhost      #IP

#
port=9991
.....
receivedFolderPath=                                # Extractor\standard_extractor\recFolder
```

启动

1. 在 “dsm_Extractor” 目录下，执行如下命令，启动采集器。

- Linux

```
# nohup ./startExtractor.sh &
```

- Windows

```
> startExtractor.bat
```

2. 启动完成后，如下图所示。

```
run extractor server...
SLF4J: Class path contains multiple SLF4J bindings.
SLF4J: Found binding in [jar:file:/home/dsm2/extractor/standard_extractor/libs/3rd/slf4j-api-1.7.25.jar/org/slf4j/impl/StaticLoggerBinder.class]
SLF4J: Found binding in [jar:file:/home/dsm2/extractor/standard_extractor/libs/3rd/slf4j-log4j12-1.7.25.jar/org/slf4j/impl/StaticLoggerBinder.class]
SLF4J: See http://www.slf4j.org/faq for an explanation.
15:19:01.240 [INFO] in ch.qos.logback.classic.LoggerContext[default] - Could NOT find resource [logback-test.xml]
15:19:01.240 [INFO] in ch.qos.logback.classic.LoggerContext[default] - Could NOT find resource [logback-test.xml]
15:19:01.240 [INFO] in ch.qos.logback.classic.LoggerContext[default] - Found resource [logback.xml] at [jar:file:/home/dsm2/extractor/standard_extractor/libs/standard/com.primeton.dps.extractor.server-7.5.jar!/logback.xml]
15:19:01.250 [INFO] in ch.qos.logback.core.joran.action.ConfigurationAction - URL [jar:file:/home/dsm2/extractor/standard_extractor/libs/standard/com.primeton.dps.extractor.server-7.5.jar!/logback.xml] is not of type file
15:19:01.385 [INFO] in ch.qos.logback.classic.joran.action.ConfigurationAction - debug attribute not set
15:19:01.388 [INFO] in ch.qos.logback.classic.joran.action.LoggerAction - Setting level of logger [org.springframework.web] to INFO
15:19:01.388 [INFO] in ch.qos.logback.core.joran.action.AppenderAction - About to instantiate appender of type [ch.qos.logback.core.ConsoleAppender]
15:19:01.390 [INFO] in ch.qos.logback.core.joran.action.AppenderAction - Naming appender as [stdout]
15:19:01.440 [INFO] in ch.qos.logback.core.joran.action.AppenderAction - About to instantiate appender of type [ch.qos.logback.core.rolling.RollingFileAppender]
15:19:01.440 [INFO] in ch.qos.logback.core.joran.action.AppenderAction - Naming appender as [applogAppender]
15:19:01.450 [INFO] in c.q.l.core.rolling.TimeBasedRollingPolicy[1517675295] - No compression will be used
15:19:01.460 [INFO] in c.q.l.core.rolling.TimeBasedRollingPolicy[1517675295] - Will use the pattern ./log/extractor-%d{yyyy-MM-dd}-%i.log for the active file
15:19:01.460 [INFO] in ch.qos.logback.core.rolling.SizeBasedFileBasedRollingPolicy[263041] - The date pattern is "yyyy-MM-dd" from file name pattern "log/extractor-%d{yyyy-MM-dd}-%i.log".
15:19:01.460 [INFO] in ch.qos.logback.core.rolling.SizeBasedFileBasedRollingPolicy[263041] - Roll over at midnight.
15:19:01.460 [INFO] in ch.qos.logback.core.rolling.SizeBasedFileBasedRollingPolicy[263041] - Setting initial period to Fri Jul 02 16:32:01 CST 2021
15:19:01.460 [WARN] in ch.qos.logback.core.rolling.SizeBasedFileBasedRollingPolicy[263041] - SizeBasedFileBasedRollingPolicy is deprecated. Use SizeBasedFileBasedRollingPolicy instead
15:19:01.460 [WARN] in ch.qos.logback.core.rolling.SizeBasedFileBasedRollingPolicy[263041] - For more information see http://logback-ops.ch/Manual/appenders.html#SizeBasedFileBasedRollingPolicy
15:19:01.470 [INFO] in ch.qos.logback.core.rolling.RollingFileAppender[applogAppender] - Active log file name: ./log/extractor.log
15:19:01.470 [INFO] in ch.qos.logback.core.rolling.RollingFileAppender[applogAppender] - File property is set to ./log/extractor.log
15:19:01.471 [INFO] in ch.qos.logback.classic.joran.action.RootLoggerAction - Setting level of ROOT logger to INFO
15:19:01.471 [INFO] in ch.qos.logback.classic.joran.action.AppenderRefAction - Attaching appender named [stdout] to Logger[ROOT]
15:19:01.471 [INFO] in ch.qos.logback.classic.joran.action.AppenderRefAction - Attaching appender named [applogAppender] to Logger[ROOT]
15:19:01.471 [INFO] in ch.qos.logback.classic.joran.action.ConfigurationAction - End of configuration.
15:19:01.472 [INFO] in ch.qos.logback.classic.joran.action.ConfigurationAction - Registering current configuration as safe fallback point
SLF4J: Actual binding is of type [ch.qos.logback.classic.util.ContextSelectorStaticBinder]
2021-07-05 16:19:01.485 [main] INFO c.p.dps.extractor.server.control.RMServerStarter - read config file: /home/dsm2/extractor/standard_extractor/conf/extractor/extractor.properties.
2021-07-05 16:19:01.493 [main] INFO c.p.dps.extractor.server.control.RMServerStarter - start extractor 132.168.38.142:9993
2021-07-05 16:19:02.421 [main] INFO com.primeton.dps.kernel.common.remote.GrpcEngine - RPC Server has started, listening on 132.168.38.142:9993
采集器标准服务运行中 ...
| 停止采集模块,请输入命令: kill -9 ps -ef|grep RMServerStarter|awk '{print $2}' |
```

卸载

直接删除 “dsm_Extractor” 目录来卸载。

2.5 数据质量管理平台安装部署前置条件

- 已配置Redis服务。
- 已下载数据质量管理平台安装介质。

软件包名称	软件包描述
Primeton_QualityCube7.0LA_Executor.tar	数据质量检核执行器安装包
Primeton_QualityCube7.0LA_Scheduler.tar	调度服务安装包
Primeton_QualityCube7.0LA_Server.tar	数据质量服务DQMS安装包
Primeton_QualityCube7.0LA_UI.tar	数据质量前端安装包

安装须知

数据质量安装介质中，数据质量服务安装包（Primeton_QualityCube7.0LA_Server.tar）、数据质量前端安装包（Primeton_QualityCube7.0LA_UI.tar）需要部署在同一台服务器，否则会导致质量分析报告页面图形无法显示。

2.5.1 检核执行器部署

安装

1. 在dqms_executor目录下解压安装包。

- Linux

```
#tar -xvf Primeton_QualityCube_7.0LA_Executor.tar -C ./dqms_executor
```

- Windows

```
用解压工具将 “Primeton_QualityCube_7.0LA_Executor.tar” 解压至 “dqms_executor” 目录
```

2. 将数据库驱动包拷贝到 “dqms_executor\lib” 目录中。

3. 修改dqms_executor/conf目录下application.yml文件。

说明

配置文件中需要修改的地方在右侧使用 “###-” 说明。其他不做修改。

```
server:
  port: 18185          ###-

mybatis:
  #
  type-aliases-package: com.primeton.**.entity
  # mapperBase
  mapper-locations: classpath:mapper/**/*.xml
  configuration:
    call-setters-on-nulls: true

common:
  jdbc:
    driverClass: com.mysql.jdbc.Driver
    username: "root"          ###-
    password: "123456"        ###-
  #
  base_dir: ${app.home:.}
  # DQMS
  damp_dir: "/root/software/dqms/Primeton_QualityCube_7.0.0_Server"  ###-Primeton_QualityCube_7.
0LA_Server

logging:
  #
  level.com.primeton.damp: INFO
  level.com.primeton.dqms: INFO

mapper:
  # \u591A\u4E2A\u63A5\u53E3\u65F6\u9017\u53F7\u9694\u5F00
  mappers: com.primeton.damp.core.mapperBase.MyMapper
  # insert \u548C update \u4E2D\uFF0C\u5224\u65AD\u5B57\u7B26\u4E32\u7C7B\u578B != ''
  not-empty: false
  # \u53D6\u56DE\u4E3B\u952E\u7684\u65B9\u5F0F
  # identity: MYSQL
  identity: MYSQL

db-type:
  driver-map: "{ 'com.mysql.cj.jdbc.Driver': 'mysql', 'com.mysql.jdbc.Driver': 'mysql', 'oracle.jdbc.
OracleDriver': 'oracle', 'com.huawei.gauss.jdbc.ZenithDriver': 'mysql', 'org.postgresql.Driver': 'postgresql',
'org.apache.hive.jdbc.HiveDriver': 'mysql' }"

spring:
  # \u6570\u636E\u6E90\uFF08\u5E94\u8BE5\u5168\u90E8\u52A0\u5BC6\uFF09
  datasource:
    #
    # \u8FDE\u63A5\uFF0C\u6CE8\u610F\u5404\u4E2A\u914D\u7F6E\uFF0C\u5C24\u5176\u662F\u8981\u4E00\u6B21\u6027\u88
```

```

4C\u591A\u6761 SQL \u65F6\uFF0C\u8981 allowMultiQueries=true
# url: jdbc:mysql://localhost:3306/damp?
useUnicode=true&useSSL=false&useLegacyDatetimeCode=false&allowMultiQueries=true&characterEncoding=utf-
8&serverTimezone=UTC
# # \u7528\u6237\u540D root
# username: damp
# #({cSs3wYoZ0BTijYqdYVj9xg==})
# # \u5BC6\u7801 root
# password: damp
# #({eC0S8Sk9b/kwt2FK0QFA9g==})
# # \u9A71\u52A8\u7C7B
# driver-class-name: com.mysql.cj.jdbc.Driver
#
\u8FDE\u63A5\uFF0C\u6CE8\u610F\u5404\u4E2A\u914D\u7F6E\uFF0C\u5C24\u5176\u662F\u8981\u4E00\u6B21\u6027\u88
4C\u591A\u6761 SQL \u65F6\uFF0C\u8981 allowMultiQueries=true
# url: jdbc:oracle:thin:@localhost:1521:orcl?
useUnicode=true&useSSL=false&useLegacyDatetimeCode=false&allowMultiQueries=true&characterEncoding=utf-
8&serverTimezone=UTC
coframe:
# \u591A\u6570\u636E\u6E90\u65F6\u8981\u4F7F\u7528jdbc-url\uFF0C\u800C\u4E0D\u662Furl
url: jdbc:mysql://192.168.30.142:3306/coframe6?useUnicode=true&characterEncoding=utf-8
###-
# \u7528\u6237\u540D root
username: ${common.jdbc.username}
#({cSs3wYoZ0BTijYqdYVj9xg==})
# \u5BC6\u7801 root
password: ${common.jdbc.password}
#({eC0S8Sk9b/kwt2FK0QFA9g==})
# \u9A71\u52A8\u7C7B
# driver-class-name: com.mysql.cj.jdbc.Driver
#driver-class-name: oracle.jdbc.OracleDriver
driver-class-name: ${common.jdbc.driverClass}
alias: coframe6
###-
#
initial-size: 10
#
min-idle: 10
#
max-active: 20
damp:
# \u591A\u6570\u636E\u6E90\u65F6\u8981\u4F7F\u7528jdbc-url\uFF0C\u800C\u4E0D\u662Furl
#url: jdbc:oracle:thin:@192.168.5.101:1521:orcl
url: jdbc:mysql://192.168.30.142:3306/coframe6?useUnicode=true&characterEncoding=UTF-
8&useSSL=false
###-
# \u7528\u6237\u540D root
username: ${common.jdbc.username}
#({cSs3wYoZ0BTijYqdYVj9xg==})
# \u5BC6\u7801 root
password: ${common.jdbc.password}

#({eC0S8Sk9b/kwt2FK0QFA9g==})
# \u9A71\u52A8\u7C7B
# driver-class-name: com.mysql.cj.jdbc.Driver
#driver-class-name: oracle.jdbc.OracleDriver
driver-class-name: ${common.jdbc.driverClass}
alias: coframe6
###-
#
initial-size: 10
#
min-idle: 10
#
max-active: 20
dqms:
# \u591A\u6570\u636E\u6E90\u65F6\u8981\u4F7F\u7528jdbc-ur\uFF0C\u800C\u4E0D\u662Furl
#url: jdbc:oracle:thin:@192.168.5.101:1521:orcl
url: jdbc:mysql://192.168.30.142:3306/coframe6?useUnicode=true&characterEncoding=utf-
8
###-
# \u7528\u6237\u540D root
username: ${common.jdbc.username}
#({cSs3wYoZ0BTijYqdYVj9xg==})
# \u5BC6\u7801 root

```

```

password: ${common.jdbc.password}
#{eC0S8Sk9b/kWt2FK0QFA9g==})
# \u9A71\u52A8\u7C7B
# driver-class-name: com.mysql.cj.jdbc.Driver
#driver-class-name: oracle.jdbc.OracleDriver
driver-class-name: ${common.jdbc.driverClass}
alias: coframe6
###-
#
initial-size: 10
#
min-idle: 10
#
max-active: 20
druid:
#
max-wait: 5000
#
time-between-connect-error-millis: 10000

# \u5206\u9875\u63D2\u4EF6
# https://github.com/pagehelper/Mybatis-PageHelper/blob/master/wikis/zh/HowToUse.md
pagehelper:
# pageSize=0 \u65F6\u67E5\u51FA\u6240\u6709\u7ED3\u679C\uFF0C\u76F8\u5F53\u4E8E\u6CA1\u5206\u9875
page-size-zero: true
# \u6570\u636E\u5E93\u65B9\u8A00
helperDialect: mysql
# helperDialect: oracle
# \u5206\u9875\u5408\u7406\u5316
# pageNum <= 0 \u65F6\u4F1A\u67E5\u8BE2\u7B2C\u4E00\u9875ssll
# pageNum > pages\uFF08\u8D85\u8FC7\u603B\u6570\u65F6\uFF0C\u4F1A\u67E5\u8BE2\u6700\u540E\u4E00\u9875
reasonable: true
# \u652F\u6301\u901A\u8FC7 Mapper \u63A5\u53E3\u53C2\u6570\u6765\u4F20\u9012\u5206\u9875\u53C2\u6570
supportMethodsArguments: true

# xxl-job
xxl:
job:
admin:
addresses: http://192.168.30.142:28282/job-admin/
username: admin
password: 123456
executor:
appname: cdh-executor
ip: 192.168.30.142
port: 9944
logpath: ${common.base_dir}/logs
logretentiondays: -1
accessToken:
###-xxl-job
###-ip

hadoop:
# hadoop HDFScluster, local
hdfs_type: local
tenant_name: ""
core-site: "${common.base_dir}/conf/core-site.xml"
hdfs-site: "${common.base_dir}/conf/hdfs-site.xml"
user_keytab: "${common.base_dir}/conf/user.keytab"
krb5_file: "${common.base_dir}/conf/krb5.conf"
# HDFS
fileDownloadUrl: "http://192.168.30.140:50070"

datashard:
# localftpsftphdfs
type: local
base_path: ${common.base_dir}/upload
#hadoop
delSrc: true
#
fileSuffix: .bcp

ftp:

```

```

host: localhost
port: 21
userName: admin
password: primeton000000

sftp:
# ip
host: localhost
#
port: 22
#
userName: root
#
password: primeton

datax:
conf:
path: ${common.base_dir}/datax/datax
remote:
#0 1 server
model: 1
server:
host: localhost
port: 22
userName: root
# 60
password: "000000"
setting:
channel: 5

datacheck:
conf:
#
corePoolSize: 10
#
maxPoolSize: 50
#
queueCapacity: 300
#
keepAliveSeconds: 1000
#conds: 1000
checkResult:
# db hive
checkFileLocalTempDir: ${common.damp_dir}/dataCheck
# ftphadopp db hive
checkFileUploadRootDir: ${common.damp_dir}/dataCheck
syncCheck:
#
checkFileLocalTempDir: ${common.base_dir}/syncCheckFile
# ftphadopp
checkFileUploadRootDir: ${common.base_dir}/syncCheckFile
sprakCheck:
#spark master
sparkMaster: spark://localhost:7077
#spark jar spark
appJarPath: /opt/spark-2.4.6-bin-hadoop2.7/appjar/sparkTask.jar
sparkTaskClass: com.primeton.sparkDataCheck.task.SparkDataCheckMainByParams
#
otherErrorDataFilePath: ${common.damp_dir}/dataCheck/otherErrorData
#
dataSummaryFilePath: ${common.damp_dir}/dataCheck/dataSummary
#
errorDetailFilePath: ${common.damp_dir}/dataCheck/errorDetail
spark:
parallel: 5
master: local[3]
deployMode: local
appName: HelloTest
jvmOpts: --driver-memory 2G --num-executors 2 --executor-cores 1
queue: default
#

```

```

files: file
# Java/Scala Spark
jars: context-1.18.0.jar,
# pyspark
pys: "${common.base_dir}/conf/pyamc.zip"
sparkSubmitOpts: -server
sparkParameters: spark.executor.userClassPathFirst=true,spark.driver.userClassPathFirst=true
appParameters:
sparkLibDir: ${common.base_dir}/spark
taskLogDir: ${common.base_dir}/logs/taskLog

kafka:
  zookeeper:
    zkList: 192.168.5.140:2181

# swagger.enable Swagger false
swagger:
  enable: false
  group: cdhexecutor
  title: CDH
  description: CDH|cdh-executor
  version: 7.0.0
  creator: primeton
  basepackage: com.primeton

coframe:
  cache:
    enabled: true
    type: memory
  auth: #
    token-expired: 7200 # token
    token-refresh-interval: 1200 # token
    relogin-keep-token: false # , token, false
    exclude-paths: "**/login,/external/**, /rest/**, /console/**" # url, .
  bean-enabled: # bean
    hibernate5-module: true # Hibernate5Module, true
    multipart-config-factory: true # MultipartConfigElement, true
    swagger-docket: ${swagger.enable:false} # coframe swagger Docket, true
    local-validator-factory-bean: true # LocalValidatorFactoryBean, true
    exception-handler-all: true # coframe, , true
  flag:
    employee-user-cascade-delete: false # ,
  filter: #
    token: # token
    enabled: false # sdktoken, true. coframe-coreJWTFilter, false
  tools:
    enabled: true # . , , . true, false

```

主要配置说明如下：其他参数使用默认配置即可。

参数	说明
server: port: 18185	
jdbc: driverClass: com.mysql.jdbc.Driver username: "root" password: "123456"	
# DAMP damp_dir: "/root/software/dqms/Primeton_QualityCube_7.0.0_Server"	Primeton_QualityCube_7.0LA_Server

<code>coframe/damp/dqms/: url: jdbc:mysql://192.168.30.142:3306/coframe6? useUnicode=true&characterEncoding=utf-8 username: \${common.jdbc.username} password: \${common.jdbc.password} driver-class-name: \${common.jdbc.driverClass} alias: coframe6</code>	<code>coframe/damp/dqms/URLalias</code>
<code>xxl: job: admin: addresses: http://192.168.30.142:28282/job-admin/ username: admin password: 123456</code>	<code>xxl-job</code>
<code>executor: appname: cdh-executor ip: 192.168.30.142 port: 9944</code>	

启动

1. 完成修改配置启动，日志在logs目录下查看。

- Linux

```
# ./startDampExecutor.sh
```

- Windows

```
> startDampExecutor.cmd
```

卸载

1. 在“dqms_executor”目录下，执行如下命令，停止数据资产目录。

- Linux

```
# ./stopDampExecutor.sh
```

- Windows

```
> stopDampExecutor.cmd
```

2. 停止服务后，直接删除“dqms_executor”目录来卸载。

2.5.2 调度服务部署

安装

1. 在dqms_scheduler目录下解压安装包。

- Linux

```
#tar -xvf Primeton_QualityCube_7.0LA_Scheduler.tar -C ./dqms_scheduler
```

- Windows

```
用解压工具将 “Primeton_QualityCube_7.0LA_Scheduler.tar” 解压至 “dqms_scheduler” 目录
```

2. 修改dqms_scheduler/config目录下application.properties文件。

说明

 配置文件需修改内容使用“###-”在右侧说明。其他默认不做修改。

```

### web
server.port=28282                #
server.servlet.context-path=/job-admin

### actuator
management.server.servlet.context-path=/actuator
management.health.mail.enabled=false

### resources
spring.mvc.servlet.load-on-startup=0
spring.mvc.static-path-pattern=/static/**
spring.resources.static-locations=classpath:/static/

### freemarker
spring.freemarker.templateLoaderPath=classpath:/templates/
spring.freemarker.suffix=.ftl
spring.freemarker.charset=UTF-8
spring.freemarker.request-context-attribute=request
spring.freemarker.settings.number_format=0.#####

### mybatis
mybatis.mapper-locations=classpath:/mybatis-mapper/*Mapper.xml
#mybatis.type-aliases-package=com.xx1.job.admin.core.model

### xxl-job, datasource
spring.datasource.url=jdbc:mysql://192.168.30.142:3306/coframe6?useUnicode=true&characterEncoding=UTF-8&autoReconnect=true&serverTimezone=Asia/Shanghai    ###-
spring.datasource.username=root                                ###-
spring.datasource.password=123456                                ###-
spring.datasource.driver-class-name=com.mysql.cj.jdbc.Driver

#spring.datasource.url=jdbc:oracle:thin:@localhost:1521:orcl
#spring.datasource.username=xxl_job
#spring.datasource.password=xxl_job
#spring.datasource.driver-class-name=oracle.jdbc.driver.OracleDriver

### datasource-pool
spring.datasource.type=com.zaxxer.hikari.HikariDataSource
spring.datasource.hikari.minimum-idle=10
spring.datasource.hikari.maximum-pool-size=30
spring.datasource.hikari.auto-commit=true
spring.datasource.hikari.idle-timeout=30000
spring.datasource.hikari.pool-name=HikariCP
spring.datasource.hikari.max-lifetime=900000
spring.datasource.hikari.connection-timeout=10000
spring.datasource.hikari.connection-test-query=SELECT 1

### xxl-job, email
spring.mail.host=smtp.qq.com
spring.mail.port=25
spring.mail.username=xxx@qq.com
spring.mail.password=xxx
spring.mail.properties.mail.smtp.auth=true
spring.mail.properties.mail.smtp.starttls.enable=true
spring.mail.properties.mail.smtp.starttls.required=true
spring.mail.properties.mail.smtp.socketFactory.class=javax.net.ssl.SSLSocketFactory

### xxl-job, access token
xxl.job.accessToken=

### xxl-job, i18n (default is zh_CN, and you can choose "zh_CN", "zh_TC" and "en")
xxl.job.i18n=zh_CN

## xxl-job, triggerpool max size
xxl.job.triggerpool.fast.max=200
xxl.job.triggerpool.slow.max=100

### xxl-job, log retention days
xxl.job.logretentiondays=30

```

主要配置说明如下：其他参数使用默认配置即可。

参数	说明
server.port=28282	
spring.datasource.url=jdbc:mysql://192.168.30.142:3306/coframe6?useUnicode=true&....() spring.datasource.username=root spring.datasource.password=123456	xxl-job.sql

启动

1. 完成修改配置启动，日志在logs目录下查看，启动服务命令如下：

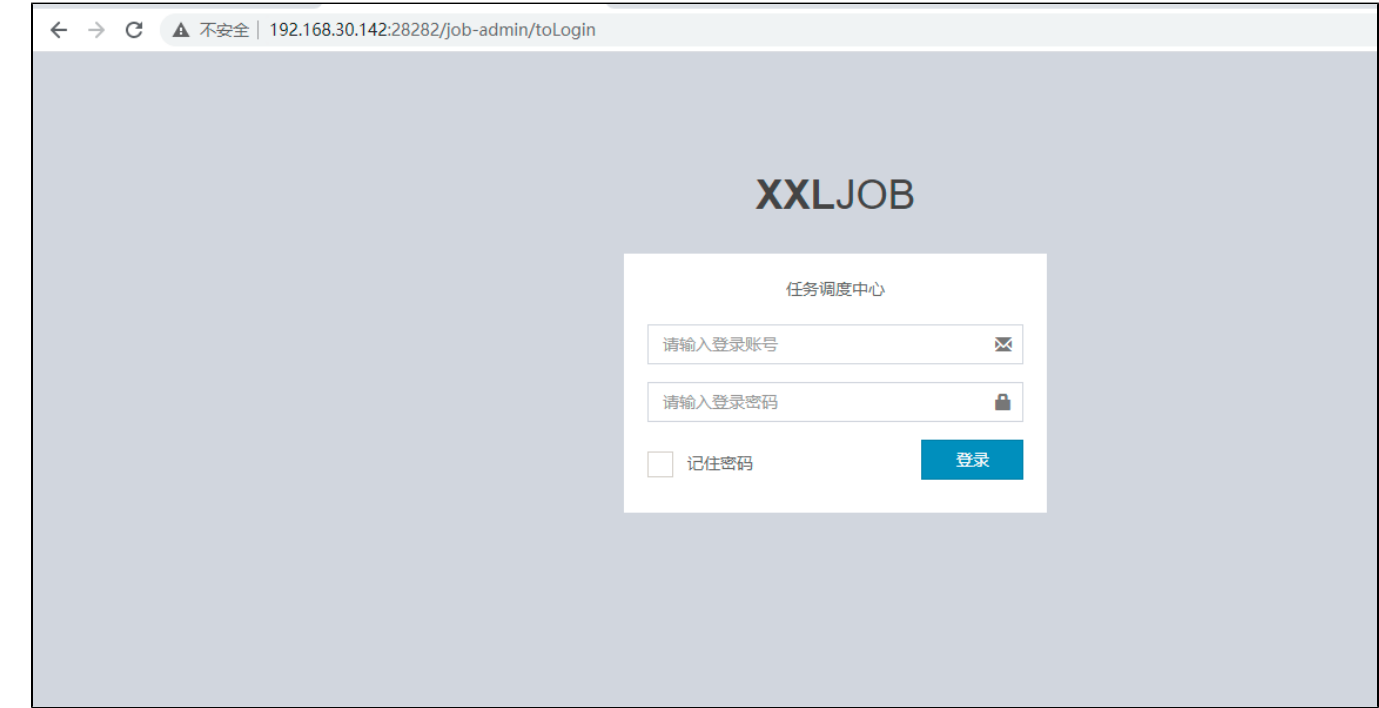
- Linux

```
# ./startJob.sh
```

- Windows

```
> startJob.cmd
```

2. 启动之后，验证如下图所示。



卸载

1. 在“dqms_scheduler”目录下，执行如下命令，停止数据服务发布。

- Linux

```
# ./stopJob.sh
```

- Windows

```
> stopJob.bat
```

2. 停止服务后，直接删除“dqms_scheduler”目录来卸载。

2.5.3 后端Server部署

配置服务

1. 创建“dqms_server”目录，解压安装包到“dqms_server”目录中。

- Linux

```
#tar -xvf Primeton_QualityCube_7.0LA_Server.tar -C ./dqms_server
```

- Windows

用解压工具将“Primeton_QualityCube_7.0LA_Server.tar”解压至“dqms_server”目录

2. 将数据库驱动包拷贝到“dqms_server\lib”目录中。

3. 在dqms_server/conf目录下修改application-prod.yml文件。

说明

配置文件中需要修改的地方在右侧使用“###-”备注说明。其他默认可以不做修改。

```
server:
  port: 8082          ###-
  servlet:
    context-path: /
    context-parameters:
      BIRT_VIEWER_LOCALE: en-US
      BIRT_VIEWER_TIMEZONE:
      WORKING_FOLDER_ACCESS_ONLY: true
      URL_REPORT_PATH_POLICY: domain
      BIRT_VIEWER_WORKING_FOLDER: /root/software/dqms/birt-directory
      BIRT_VIEWER_DOCUMENT_FOLDER: report-engine/documents
      BIRT_VIEWER_IMAGE_DIR: report-engine/images
      BIRT_VIEWER_LOG_DIR: report-engine/logs
      BIRT_VIEWER_SCRIPTLIB_DIR: report-engine/scriptlib
      BIRT_VIEWER_LOG_LEVEL: WARNING
      BIRT_RESOURCE_PATH:
      BIRT_VIEWER_MAX_ROWS:
      BIRT_VIEWER_MAX_CUBE_ROWLEVELS:
      BIRT_VIEWER_MAX_CUBE_COLUMNLEVELS:
      BIRT_VIEWER_CUBE_MEMORY_SIZE:
      BIRT_VIEWER_CONFIG_FILE: /WEB-INF/viewer.properties
      BIRT_VIEWER_PRINT_SERVERSIDE: ON
      HTML_ENABLE_AGENTSTYLE_ENGINE: true
      BIRT_FILENAME_GENERATOR_CLASS: org.eclipse.birt.report.utility.filename.DefaultFilenameGenerator
#  ssl:
#    key-store: classpath:tomcat.keystore
#    key-store-password: 123456
#    key-store-type: JKS
#    key-alias: server

common:
  jdbc:
    url: "jdbc:mysql://localhost:3306/db_name?useUnicode=true&characterEncoding=utf-8" ###-
    driverClass: com.mysql.jdbc.Driver
    host: ""
    username: "root"          ###-
    password: "123456"        ###-
    alias: "db_name"          ###- db_name
    base_dir: ${app.home:.}

spring:
  redis:
    host: 192.168.30.144      ###-redisip,
    port: 6379                ###-redis
    password: primeton000000  ###-redis
  freemarker:
    cache: false             #
#
datasource:
```

```

# # SQL allowMultiQueries=true
# url: jdbc:mysql://localhost:3306/damp?
useUnicode=true&useSSL=false&useLegacyDatetimeCode=false&allowMultiQueries=true&characterEncoding=utf-
8&serverTimezone=UTC
# # root
# username: damp
# #({cSs3wYoZ0BTijYqdYVj9xg==})
# # root
# password: damp
# #({eC0S8Sk9b/kwt2FK0QFA9g==})
# #
# driver-class-name: com.mysql.cj.jdbc.Driver
# SQL allowMultiQueries=true
# url: jdbc:oracle:thin:@localhost:1521:orcl?
useUnicode=true&useSSL=false&useLegacyDatetimeCode=false&allowMultiQueries=true&characterEncoding=utf-
8&serverTimezone=UTC
coframe:
# jdbc-urlurl
#url: jdbc:oracle:thin:@192.168.5.101:1521:orcl
url: ${common.jdbc.url}
#url: jdbc:mysql://192.168.30.144:3306/coframe51?useUnicode=true&characterEncoding=utf-8
# root
username: root
#({cSs3wYoZ0BTijYqdYVj9xg==})
# root
password: ${common.jdbc.password}
#({eC0S8Sk9b/kwt2FK0QFA9g==})
#
# driver-class-name: com.mysql.cj.jdbc.Driver
#driver-class-name: oracle.jdbc.OracleDriver
driver-class-name: ${common.jdbc.driverClass}
alias: ${common.jdbc.alias}
#
initial-size: 10
#
min-idle: 10
#
max-active: 20
damp:
enabled: true
# jdbc-urlurl
#url: jdbc:oracle:thin:@192.168.5.101:1521:orcl
url: ${common.jdbc.url}
# root
username: ${common.jdbc.username}
#({cSs3wYoZ0BTijYqdYVj9xg==})
# root
password: ${common.jdbc.password}

#({eC0S8Sk9b/kwt2FK0QFA9g==})
#
# driver-class-name: com.mysql.cj.jdbc.Driver
#driver-class-name: oracle.jdbc.OracleDriver
driver-class-name: ${common.jdbc.driverClass}
alias: ${common.jdbc.alias}
#
initial-size: 10
#
min-idle: 10
#
max-active: 20
dqms:
enabled: true
# jdbc-urlurl
#url: jdbc:oracle:thin:@192.168.5.101:1521:orcl
url: ${common.jdbc.url}
# root
username: ${common.jdbc.username}
#({cSs3wYoZ0BTijYqdYVj9xg==})
# root
password: ${common.jdbc.password}

```

```

#({eCOS8Sk9b/kwt2FK0QFA9g==})
#
# driver-class-name: com.mysql.cj.jdbc.Driver
#driver-class-name: oracle.jdbc.OracleDriver
driver-class-name: ${common.jdbc.driverClass}
alias: ${common.jdbc.alias}
#
initial-size: 10
#
min-idle: 10
#
max-active: 20
metacube:
enabled: true
# jdbc-urlurl
#url: jdbc:oracle:thin:@127.0.0.1:1521:orcl
url: ${common.jdbc.
url}
###-
# root
username: ${common.jdbc.
username}
#({cSs3wYoZ0BTijYqdYVj9xg==})
# root
password:
"primeton000000"
###-
#({eCOS8Sk9b/kwt2FK0QFA9g==})
#
driver-class-name: ${common.jdbc.driverClass}
#driver-class-name: oracle.jdbc.OracleDriver
#schema
alias: ${common.jdbc.
alias}
###-db_name
#
initial-size: 2
#
min-idle: 2
#
max-active: 5
xxl-job:
enabled: true
# jdbc-urlurl
#url: jdbc:oracle:thin:@127.0.0.1:1521:orcl
url: ${common.jdbc.url}
# root
username: ${common.jdbc.username}
#({cSs3wYoZ0BTijYqdYVj9xg==})
# root
password: ${common.jdbc.password}
#({eCOS8Sk9b/kwt2FK0QFA9g==})
#
driver-class-name: ${common.jdbc.driverClass}
#driver-class-name: oracle.jdbc.OracleDriver
#schema
alias: ${common.jdbc.alias}
#
initial-size: 3
#
min-idle: 3
#
max-active: 8
druid:
#
max-wait: 5000
#
time-between-connect-error-millis: 10000

jpa:

```

```

    open-in-view: false
cache:
  #
  type: redis
  redis:
    # key
    key-prefix: dampDev
    #
    time-to-live: 36000s

servlet:
  # http://ip:port/files/xxx.txt
  staticAccessPath: /files/**
  #uploadPath: /home/damp/uploadFiles/
  uploadPath: ${common.base_dir}/upload/principal
  multipart:
    # ( true)
    enabled: true
    # realfile.max-size 5000M 1M springboot
    max-file-size: 5000MB
    # 5000M10M springboot
    max-request-size: 5000MB
  # solr
solr:
  solrUser: solr
  solrPassword: solr!123
  serverUrl: http://192.168.30.142:17000/solr
  collectionName: dgp
  #
  commitLimit: 5000
  #
  copyLimit: 500
  #
  threadNum: 20

logging:
  #
  level.com.primeton.damp: INFO
  level.com.primeton.dqms: INFO

# Json web token
jwt:
  #
  expire-time: 600m
  # claim key
  claim-key-auth: auth
  # key
  header: Authorization
  # token
  token-type: Bearer

xxl:
  job:
    admin:
      addresses: http://192.168.30.142:28282/job-admin/
      username: admin
      password: 123456
      #xxl-job

hadoop:
  # hadoop HDFScluster, local
  hdfs_type: local
  tenent_name: ""
  core-site: "${common.base_dir}/conf/core-site.xml"
  hdfs-site: "${common.base_dir}/conf/hdfs-site.xml"
  user_keytab: "${common.base_dir}/conf/user.keytab"
  krb5_file: "${common.base_dir}/conf/krb5.conf"
  # HDFS
  fileDownloadUrl: "http://192.168.30.140:50070"

cache:
  type: mem

```

```

datashard:
  # localftpsftphdfs
  type: local
  base_path: ${common.base_dir}/upload
  #hadoop
  delSrc: true
  #
  fileSuffix: .bcp

ftp:
  host: localhost
  port: 21
  userName: admin
  password: primeton000000

sftp:
  # ip
  host: localhost
  #
  port: 22
  #
  userName: root
  #
  password: primeton

file:
  dqms:
    #
    otherErrorDataFilePath: ${common.base_dir}/dataCheck/otherErrorData
    #
    dataSummaryFilePath: ${common.base_dir}/dataCheck/dataSummary
    #
    errorDetailFilePath: ${common.base_dir}/dataCheck/errorDetail
  damp:
    #
    #dataTemplateFilePath: /home/resourceTemplate/dataResourceImport.xlsx
    dataTemplateFilePath: ${common.base_dir}/resourceTemplate/resourceTemplate.xlsx
    # 5M
    templateFileLocation: ${common.base_dir}/fileTemplate/dataResourceImport.xlsx
    #
    VerificationTemplatePath: ${common.base_dir}/resourceTemplate/VerificationTemplate.xlsx
    clclassTemplateFilePath: ${common.base_dir}/resourceTemplate/classifDataImport.xlsx

#
metacube:
  address: http://192.168.30.143:10800/dgs

datacheck:
  conf:
    #
    corePoolSize: 10
    #
    maxPoolSize: 50
    #
    queueCapacity: 300
    #
    keepAliveSeconds: 1000
  checkResult:
    # apache mrs cdh
    warehouseType: apache
    #
    checkFileLocalTempDir: ${common.base_dir}/dataCheck
    #(ftp)
    checkFileUploadRootDir: ${common.base_dir}/dataCheck
    #ftpHadoop
    isSyncHadoop: true
  DBkerberos:

```

```
#po_filefilepath
path: ${common.base_dir}/kerberos/
```

主要配置说明如下：其他参数使用默认配置即可。

参数	说明
port	
common: jdbc: url: "jdbc:mysql://192.168.30.144:3306/product_integration? useUnicode=true&characterEncoding=utf-8" driverClass: com.mysql.jdbc.Driver host: "" username: "root" password: "primeton000000" alias: "product_integration"	jdbcurlusernamepassword alias alias
redis: host: 192.168.30.144 port: 6379 password: primeton000000	redisip,
metacube: password: "primeton000000"	
cache: # type: redis	
xxl: job: admin: addresses: http://192.168.16.166:28282/job-admin/ username: admin password: 123456	xxl-job
metacube: address: http://localhost:10800/dgs	(ip)

3. 在dqms_server/conf目录下修改application.yml文件，内容如下：

```

# pageNum <= 0 时会查询第一页sql
# pageNum > pages (超过总数时), 会查询最后一页
reasonable: true
# 支持通过 Mapper 接口参数来传递分页参数
supportMethodsArguments: true

# swagger.enable 控制是否加载 Swagger, 如果为 false 则不加载, 否则加载
swagger:
  enable: false
  group: damp
  title: DAMP接口文档
  description: DAMP接口文档
  version: 7.1.0
  creator: primeton
  basepackage: com.primeton.damp

coframe:
  cache:
    enabled: true
    type: redis # memory, redis 修改type值为, redis
    key-namespace: integration167 # 修改key-namespace值和其他特集成应用的rediskey-namespace值一致, 如, integration167
  auth: # 认证配置
    token-expired: 7200 # token超时时间
    token-refresh-interval: 1200 # token刷新时间
    relogin-keep-token: false # 重复登陆时, 是否使用上次登陆产生的token, 默认为false
    exclude-paths: /images/**,/fonts/**,/js/**,**/login,/druid/**,/rest/**,/console/**,*.js,*.do,*.css,*.png,*.gif,*.jpg,*.jpeg # 不需要进行权限校验的url, 这里可以将静态资源排除, 多个路径之间用逗号连接
  bean-enabled: # 是否启用bean
  hibernate5-module: true # 是否启用 HibernateModule, 默认为true
  multipart-config-factory: true # 是否启用 MultipartConfigElement, 默认为true
  swagger-docket: ${swagger.enable:false} # 是否启用 coframe 的 swagger Docket, 默认为true
  local-validator-factory-bean: true # 是否启用 LocalValidatorFactoryBean, 默认为true
  exception-handler-all: true # 是否启用coframe的统一异常处理, 处理所有异常类, 默认为true
  flag:
    employee-user-cascade-delete: false # 删除员工时, 是否级连删除用户
  filter: # 过滤器配置
    token: # token过滤器
      enabled: false # 是否启用sdk中的token校验过滤器, 默认为true. 因为coframe-core模块中已经有了JWTFilter, 所以这里必须设置为false
  tools:
    enabled: true # 是否向外提供工具类型的接口, 目前提供的工具接口中, 暴露了一个功能扫描的接口, 调用者可以调用此接口, 获取代码中定义的 功能定义. 此配置建议开发时设置为true, 而生产环境中设置为false

tidSkipPatterns: derbyLocale_*.jar,jaxbi-impl.jar,jsr173_1.0_api.jar,activation.jar,jaxb-api.jar,osdt_cert.jar,osdt_core.jar,shims-0.7.45.jar,oraclepki.jar,xml-apis.jar,xercesImpl.jar
-- INSERT --

```

启动

1. 完成修改配置启动, 日志在logs目录下查看。

- Linux

```
# ./startServer.sh
```

- Windows

```
> startServer.cmd
```

2. 完成启动后, 如下图所示。

```

2021-07-06 18:33:43 [main] - [ INFO ] [ s.d.s.w.r.operation.CachingOperationNameGenerator : 40 ] - Generating unique operation named: getDictItemListUsingGET_1
2021-07-06 18:33:43 [main] - [ INFO ] [ s.d.s.w.r.operation.CachingOperationNameGenerator : 40 ] - Generating unique operation named: getDictTypeByIDUsingGET_2
2021-07-06 18:33:43 [main] - [ INFO ] [ s.d.s.w.r.operation.CachingOperationNameGenerator : 40 ] - Generating unique operation named: getDictTypeListUsingGET_1
2021-07-06 18:33:43 [main] - [ INFO ] [ s.d.s.w.r.operation.CachingOperationNameGenerator : 40 ] - Generating unique operation named: updateDictTypeUsingPUT_1
2021-07-06 18:33:43 [main] - [ INFO ] [ s.d.s.w.r.operation.CachingOperationNameGenerator : 40 ] - Generating unique operation named: validateDictItemUsingGET_1
2021-07-06 18:33:43 [main] - [ INFO ] [ s.d.s.w.r.operation.CachingOperationNameGenerator : 40 ] - Generating unique operation named: validateDictTypeUsingGET_1
2021-07-06 18:33:43 [main] - [ INFO ] [ s.d.s.w.r.operation.CachingOperationNameGenerator : 40 ] - Generating unique operation named: addUsingPOST_17
2021-07-06 18:33:43 [main] - [ INFO ] [ s.d.s.w.r.operation.CachingOperationNameGenerator : 40 ] - Generating unique operation named: deleteUsingDELETE_17
2021-07-06 18:33:43 [main] - [ INFO ] [ s.d.s.w.r.operation.CachingOperationNameGenerator : 40 ] - Generating unique operation named: detailUsingGET_17
2021-07-06 18:33:43 [main] - [ INFO ] [ s.d.s.w.r.operation.CachingOperationNameGenerator : 40 ] - Generating unique operation named: listUsingGET_18
2021-07-06 18:33:43 [main] - [ INFO ] [ s.d.s.w.r.operation.CachingOperationNameGenerator : 40 ] - Generating unique operation named: updateUsingPUT_29
2021-07-06 18:33:43 [main] - [ INFO ] [ s.d.s.w.r.operation.CachingOperationNameGenerator : 40 ] - Generating unique operation named: listUsingGET_19
2021-07-06 18:33:43 [main] - [ INFO ] [ s.d.s.w.r.operation.CachingOperationNameGenerator : 40 ] - Generating unique operation named: createUsingPOST_14
2021-07-06 18:33:43 [main] - [ INFO ] [ s.d.s.w.r.operation.CachingOperationNameGenerator : 40 ] - Generating unique operation named: deleteByIdsUsingPUT_8
2021-07-06 18:33:43 [main] - [ INFO ] [ s.d.s.w.r.operation.CachingOperationNameGenerator : 40 ] - Generating unique operation named: findByIdUsingGET_14
2021-07-06 18:33:43 [main] - [ INFO ] [ s.d.s.w.r.operation.CachingOperationNameGenerator : 40 ] - Generating unique operation named: pagingByCriteriaUsingGET_7
2021-07-06 18:33:43 [main] - [ INFO ] [ s.d.s.w.r.operation.CachingOperationNameGenerator : 40 ] - Generating unique operation named: queryByCriteriaUsingGET_4
2021-07-06 18:33:43 [main] - [ INFO ] [ s.d.s.w.r.operation.CachingOperationNameGenerator : 40 ] - Generating unique operation named: updateUsingPUT_30
2021-07-06 18:33:43 [main] - [ INFO ] [ s.d.s.w.r.operation.CachingOperationNameGenerator : 40 ] - Generating unique operation named: scanFunctionsUsingGET_1
2021-07-06 18:33:43 [main] - [ INFO ] [ s.d.s.w.r.operation.CachingOperationNameGenerator : 40 ] - Generating unique operation named: createUsingPOST_15
2021-07-06 18:33:43 [main] - [ INFO ] [ s.d.s.w.r.operation.CachingOperationNameGenerator : 40 ] - Generating unique operation named: deleteByIdsUsingDELETE_6
2021-07-06 18:33:43 [main] - [ INFO ] [ s.d.s.w.r.operation.CachingOperationNameGenerator : 40 ] - Generating unique operation named: findByIdUsingGET_15
2021-07-06 18:33:43 [main] - [ INFO ] [ s.d.s.w.r.operation.CachingOperationNameGenerator : 40 ] - Generating unique operation named: pagingByCriteriaUsingGET_8
2021-07-06 18:33:43 [main] - [ INFO ] [ s.d.s.w.r.operation.CachingOperationNameGenerator : 40 ] - Generating unique operation named: updateUsingPUT_31
2021-07-06 18:33:43 [main] - [ INFO ] [ s.d.s.w.r.operation.CachingOperationNameGenerator : 40 ] - Generating unique operation named: pagingByCriteriaUsingGET_9
2021-07-06 18:33:43 [main] - [ INFO ] [ s.d.s.w.r.operation.CachingOperationNameGenerator : 40 ] - Generating unique operation named: queryByCriteriaUsingGET_5
2021-07-06 18:33:43 [main] - [ INFO ] [ s.d.s.w.r.operation.CachingOperationNameGenerator : 40 ] - Generating unique operation named: createUsingPOST_16
2021-07-06 18:33:43 [main] - [ INFO ] [ s.d.s.w.r.operation.CachingOperationNameGenerator : 40 ] - Generating unique operation named: deleteByIdsUsingDELETE_7
2021-07-06 18:33:43 [main] - [ INFO ] [ s.d.s.w.r.operation.CachingOperationNameGenerator : 40 ] - Generating unique operation named: findByIdUsingGET_16
2021-07-06 18:33:43 [main] - [ INFO ] [ s.d.s.w.r.operation.CachingOperationNameGenerator : 40 ] - Generating unique operation named: pagingByCriteriaUsingGET_10
2021-07-06 18:33:43 [main] - [ INFO ] [ s.d.s.w.r.operation.CachingOperationNameGenerator : 40 ] - Generating unique operation named: queryByCriteriaUsingGET_6
2021-07-06 18:33:43 [main] - [ INFO ] [ s.d.s.w.r.operation.CachingOperationNameGenerator : 40 ] - Generating unique operation named: querySubEmployeesUsingGET_2
2021-07-06 18:33:43 [main] - [ INFO ] [ s.d.s.w.r.operation.CachingOperationNameGenerator : 40 ] - Generating unique operation named: updateUsingPUT_32
2021-07-06 18:33:43 [main] - [ INFO ] [ o.s.boot.web.embedded.tomcat.TomcatWebServer : 202 ] - Tomcat started on port(s): 19999 (http) with context path ''
2021-07-06 18:33:43 [main] - [ INFO ] [ com.primeton.dqms.Application : 59 ] - Started Application in 20.264 seconds (JVM running for 21.049)

```

说明

启动中后台若出现如下错误, 需要在数据库连接url中加入useSSL=true或者false即可, 如下所示。

url=jdbc:mysql://127.0.0.1:3306/framework?useSSL=false

```

Fri Jul 30 18:40:50 CST 2021 WARN: Establishing SSL connection without server's identity verification is not recommended. According to MySQL 5.5.45+, 5.6.26+ and 5.7.6+ requirements SSL connection must be established by default if explicit option isn't set. For compliance with existing applications not using SSL the verifyServerCertificate property is set to 'false'. You need either to explicitly disable SSL by setting useSSL=false, or set useSSL=true and provide truststore for server certificate verification.

```

卸载

1. 在“dqms_server”目录下，执行如下命令，停止数据资产目录。

- Linux

```
# ./stopServer.sh
```

- Windows

```
> stopServer.cmd
```

2. 停止服务后，直接删除“dqms_server”目录来卸载。

2.6 Nginx配置

说明

 本示例中nginx配置数据资产目录为登录的统一入口。

前置条件

- 服务器端已安装Nginx。
- 已下载个应用的前端介质包：
 - 元数据前端介质包：Primeton_MetaCube_7.5LA_UI.tar
 - 数据资产目录前端介质包：Primeton_DataCatalogCube_7.0LA_UI.tar
 - 数据标准前端介质包：Primeton_DataStandardCube_7.0LA_UI.tar
 - 数据质量前端介质包：Primeton_QualityCube_7.0LA_UI.tar

前端UI部署

本示例默认将所有应用安装在/home/primeton下。

1. 在/home/primeton目录下分别创建 dgs_ui、damp_ui、dqms_ui、dsm_ui目录。

```
# mkdir dgs_ui damp_ui dqms_ui dsm_ui
```

2. 在分别/home/primeton目录下将各应用前端包解压在对应目录下。

```
# tar -xvf Primeton_MetaCube_7.5LA_UI.tar -C ./dgs_ui/
# tar -xvf Primeton_DataCatalogCube_7.0LA_UI.tar -C ./damp-ui
# tar -xvf Primeton_QualityCube_7.0LA_UI.tar -C ./dqms-ui
# tar -xvf Primeton_DataStandardCube_7.0LA_UI.tar -C ./dsm_ui/
```

配置Nginx

1. 查看/etc/nginx/nginx.conf配置文件内容如下,在http块下增加设置上传文件大小: client_max_body_size 20m;

```

user root;                                #root
worker_processes auto;

error_log /var/log/nginx/error.log notice;
pid /var/run/nginx.pid;

events {
worker_connections 1024;
}

http {
include /etc/nginx/mime.types;
default_type application/octet-stream;
client_max_body_size 20m;

log_format main '$remote_addr - $remote_user [$time_local] "$request" '
'$status $body_bytes_sent "$http_referer" '
'"$http_user_agent" "$http_x_forwarded_for"';

access_log /var/log/nginx/access.log main;

sendfile on;
#tcp_nopush on;

keepalive_timeout 65;

#gzip on;

include /etc/nginx/conf.d/*.conf;
}

```

2. 修改Nginx配置文件/etc/nginx/conf.d/default.conf, 参考如下:

```

# nginx
server {
listen 18789;
server_name 192.168.16.167;
charset utf-8;

location / {
charset utf-8;
root /home/primeton/damp_ui;
autoindex on;
index index.html index.htm;
}

#/damp/ /damp
location /damp/ {
charset utf-8;
index index.html index.htm;
proxy_pass http://192.168.16.167:8089/;
}

#/forward-dgs/,dgsginxforward-dgs
location /forward-dgs/ {
proxy_pass http://192.168.16.167:18780/;
}

#/dgs/ /dgs
location /dgs/ {
charset utf-8;
proxy_pass http://192.168.16.167:8080/;
}

location /forward-dsm/ {
proxy_pass http://192.168.16.167:18781/;
}

```

```

    }

    location /dsm/ {
        charset utf-8;
        proxy_pass http://192.168.16.167:8088/;
    }

    location /forward-dqms/ {
        proxy_pass http://192.168.16.167:18782/;
    }

    location /dqms/ {
        charset utf-8;
        proxy_pass http://192.168.16.167:8082/;
    }

}

#nginx
server {
    listen      18780;
    server_name 192.168.16.167;
    charset utf-8;

    location / {
        charset utf-8;
        root   /home/primeton/dgs_ui;
        index  index.html index.htm;
    }
    location /dgs/ {
        charset utf-8;
        proxy_pass http://192.168.16.167:8080/;
    }
}

server {
    listen      18781;
    server_name 192.168.16.167;
    charset utf-8;

    location / {
        charset utf-8;
        autoindex on;
        root   /home/primeton/dsm_ui;
        index  index.html index.htm;
    }
    location /dsm/ {
        charset utf-8;
        proxy_pass http://192.168.16.167:8088/;
    }
}

server {
    listen      18782;
    server_name 192.168.16.167;
    charset utf-8;

    location / {
        charset utf-8;
        root   /home/primeton/dqms_ui;
        index  index.html index.htm;
    }
    location /dqms/ {
        charset utf-8;
        proxy_pass http://192.168.16.167:8082/;
    }
}

```

```
}

```

本示例中各应用后端端口和nginx中配置的前端端口配置如下：

应用	后端IP	后端端口	Nginx前端端口	前端介质解压目录
元数据管理平台 (DGS)	192.168.16.167	8080	18780	/home/primeton/dgs_ui
数据标准 (DSM)	192.168.16.167	8088	18781	/home/primeton/dsm_ui
数据资产目录 (DAMP)	192.168.16.167	8089	18789	/home/primeton/damp_ui
数据质量 (DQMS)	192.168.16.167	8082	18782	/home/primeton/dqms_ui

启动前端服务

```
nginx

```

重新加载

```
nginx -s reload

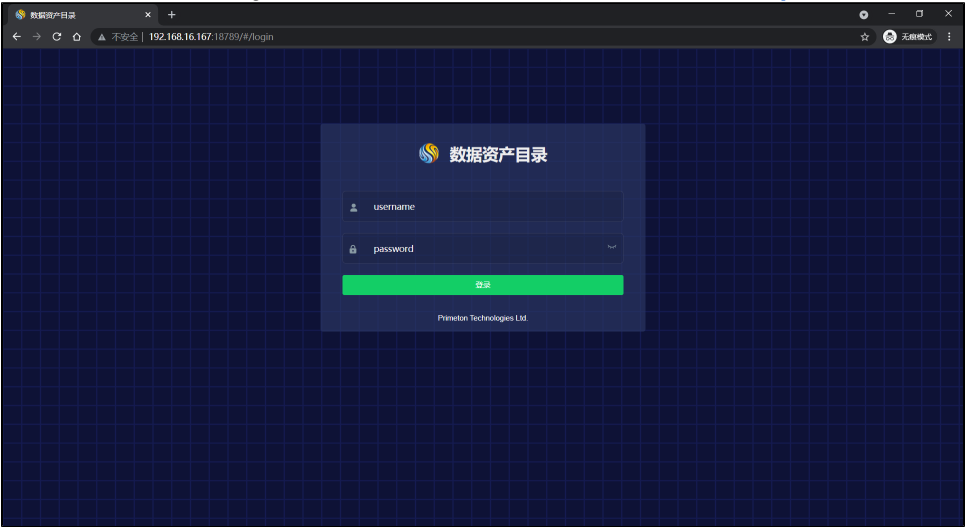
```

2.7 部署后修改参数

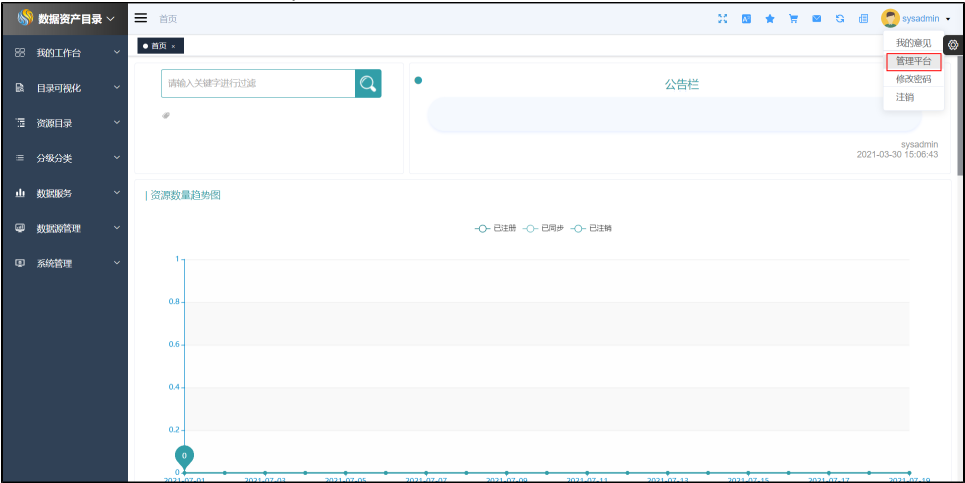
说明

成功启动后端各应用的服务，前端nginx部署成功且已启动。

1. 访问数据资产入口，端口为nginx中配置的数据资产访问端口。本示例的访问链接为：<http://192.168.16.167:18789/#/login>，如下图：



2. 使用系统默认管理员账户登录（sysadmin/000000）。



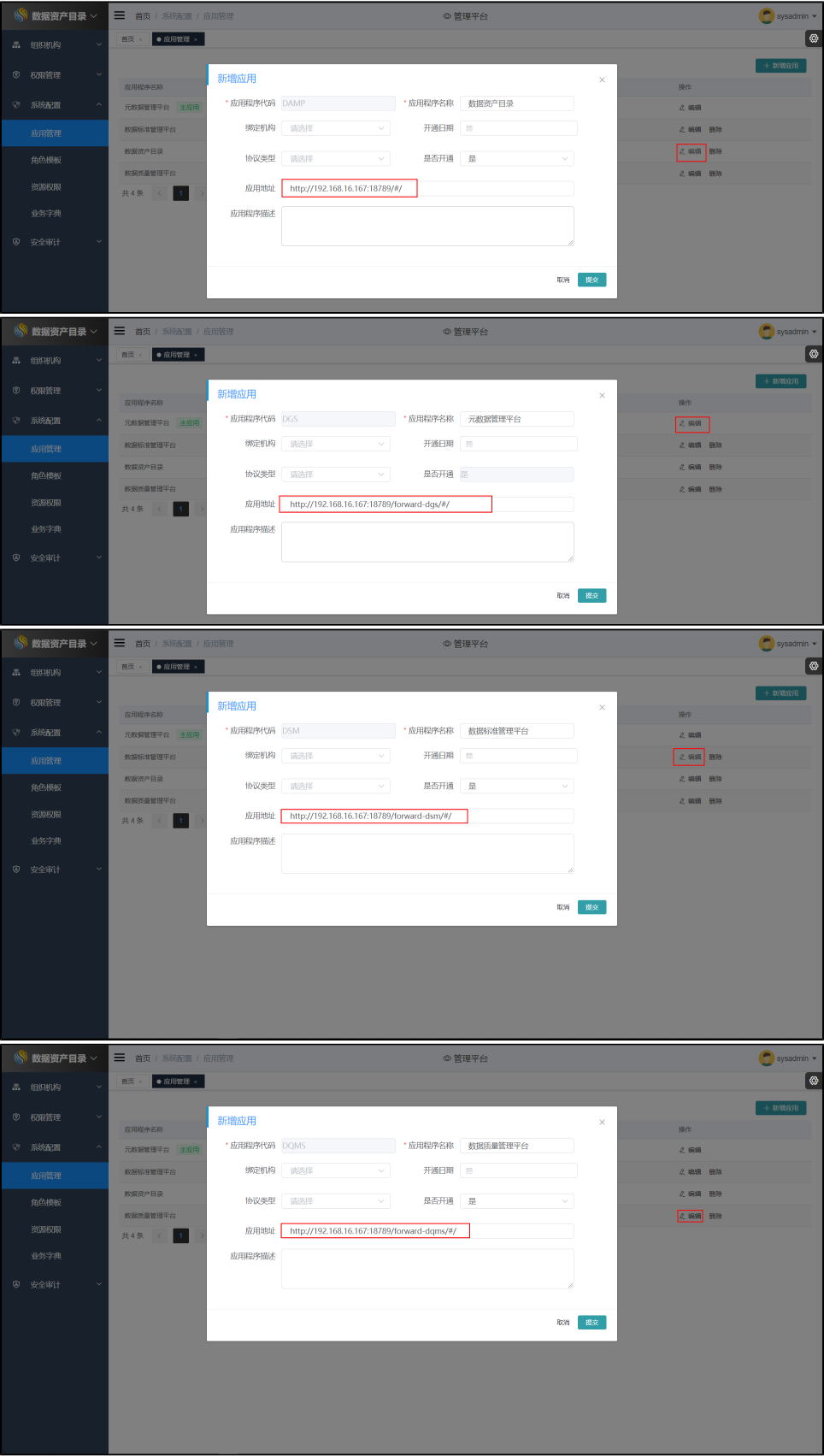
3. 点击管理平台入口，进入后台管理平台页面，默认显示的首页可修改本账号的默认应用，当前默认数据资产目录为默认应用。



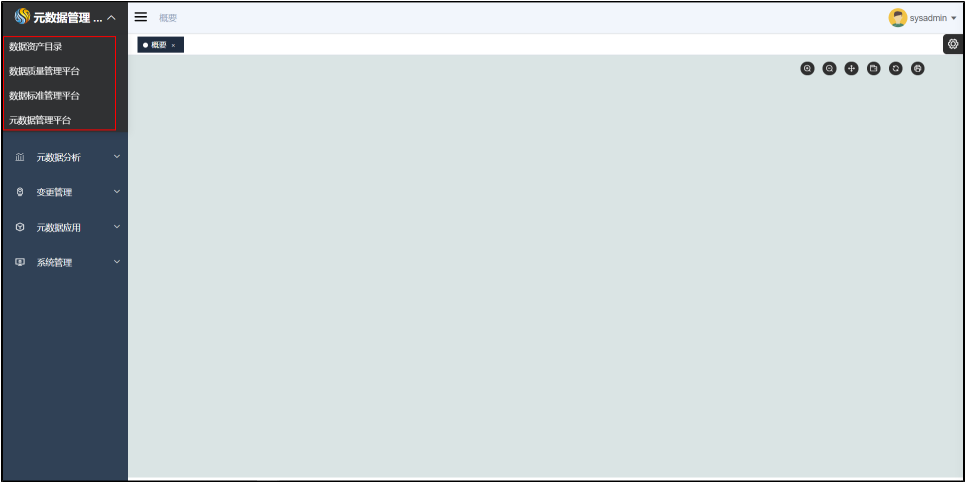
4. 点击“系统配置>应用管理”页面，编辑各应用的应用地址字段，实现同账号一次登录后在不同应用中跳转。

说明

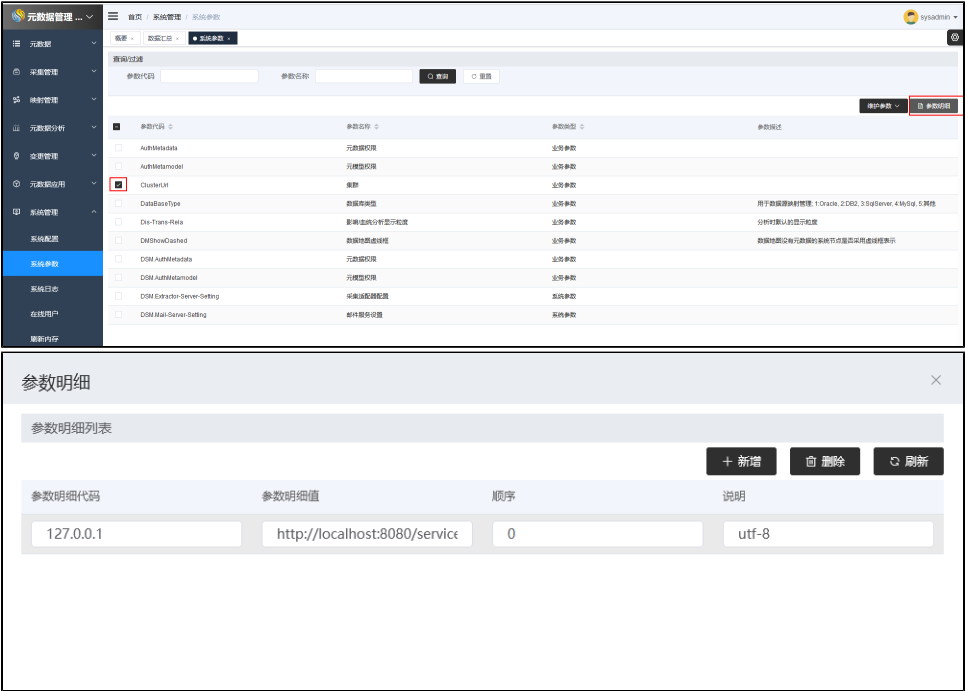
 各应用地址和nginx中配置需保持一致。统一登录入口的应用地址格式：[http://ip:主应用端口/#/](#)，其他应用地址使用代理跳转方式，应用地址格式为：[http://ip:主应用端口/代理路由/#/](#)。注意：结尾的【/#/】必须配置。
如：本示例将数据资产目录在nginx中配置为统一登录入口（[http://192.168.16.167:18789/#/](#)），元数据使用代理跳转方式（[http://192.168.16.167:18789/forward-dgs/#/](#)），配置如下图。



5. 配置完成后，重新登录应用，点击左上角的已部署的应用，可实现同账号一次登录后在不同应用中跳转。



6. 点击元数据管理平台，在“系统管理>系统参数”页面，勾选ClusterUrl，点击右侧的“参数明细”，在参数明细值页面，修改参数明细值，去掉url中的/dgs，即为：<http://localhost:8080/services/loadmemory>，输入说明值：utf-8, 点击保存。



说明
元数据管理平台，在“系统管理>数据汇总”页面，点击“执行 数据流向载入内存”，修改上述参数后可正常执行该指令。

7. 点击元数据管理平台，单击左侧导航树上的“系统管理 > 系统配置”，进入“系统配置”界面。修改“Web服务器(RMI)主机IP”为DGS系统所在的IP地址。

元数据管理平台 / 系统管理 / 系统配置

系统配置

邮件服务设置

邮件服务器主机或IP: mail.primedata.com

邮件服务器端口: 25

邮件发送协议: SMTP

邮件服务器需要身份验证: ☒ 是 ☐ 否

认证账户:

认证密码:

连接超时时间(毫秒): 25000

系统默认发件人: metamanager@primedata.com

系统访问地址: http://localhost:8080/mm

服务器目录设置

文件上传目录: /opt/MM/UPLOAD

文件下载目录: /opt/MM/DOWNLOAD

文件附件目录: /opt/MM/ATTACH

文件临时目录: /opt/MM/TEMPORARY

采集适配器配置

Web服务器(RMI)主机IP: 192.168.16.167

Web服务器(RMI)端口: 9998

8. 点击数据标准管理平台，单击左侧导航树上的“系统管理 > 系统配置”，进入“系统配置”界面。修改“Web服务器(RMI)主机IP”为DSM Server系统所在的IP地址。

数据标准管理 / 系统管理 / 系统配置

系统配置

邮件服务设置

邮件服务器主机或IP: mail.primedata.com

邮件服务器端口: 25

邮件发送协议: SMTP

邮件服务器需要身份验证: ☒ 是 ☐ 否

认证账户:

认证密码:

连接超时时间(毫秒): 25000

系统默认发件人: standardmanager@primedata.com

系统访问地址: http://localhost:8080/dsm

服务器目录设置

文件上传目录: /opt/DSM/UPLOAD

文件下载目录: /opt/DSM/DOWNLOAD

文件附件目录: /opt/DSM/ATTACH

文件临时目录: /opt/DSM/TEMPORARY

采集适配器配置

Web服务器(RMI)主机IP: 192.168.16.167

Web服务器(RMI)端口: 9996

03 快速入门指南

场景说明

本快速入门以元数据管理平台采集元数据，经数据资产目录系统同步后注册成为资产，在质量管理平台进行检核后，再由数据资产目录发布成服务对外提供。主要分为以下几部分内容：

1. 元数据管理平台：元数据采集
2. 数据资产目录：资源同步、注册
3. 数据质量管理平台：质量检核
4. 数据资产目录：服务发布

3.1 元数据采集

元数据的采集流程可参考：[数据字典采集](#)。

3.2 资源同步、注册

资源同步

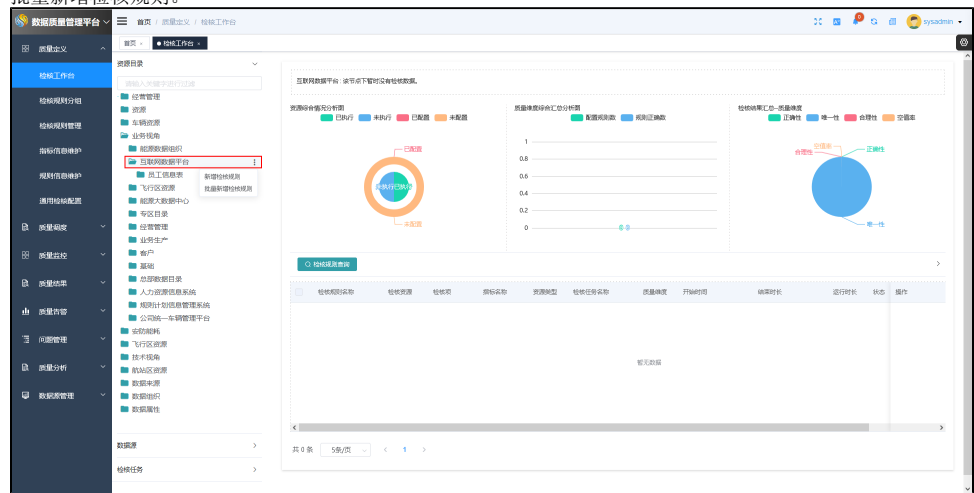
资源同步流程可参考：[数据资产同步](#)。

资源注册

资源同步流程可参考：[数据资产注册](#)。

3.3 资产数据质量检核

1. 点击数据质量管理平台的“质量定义>检核工作台”页面，在页面显示资源目录树，点击发布的资源所在目录，目录下拉按钮显示：新增检核规则、批量新增检核规则。



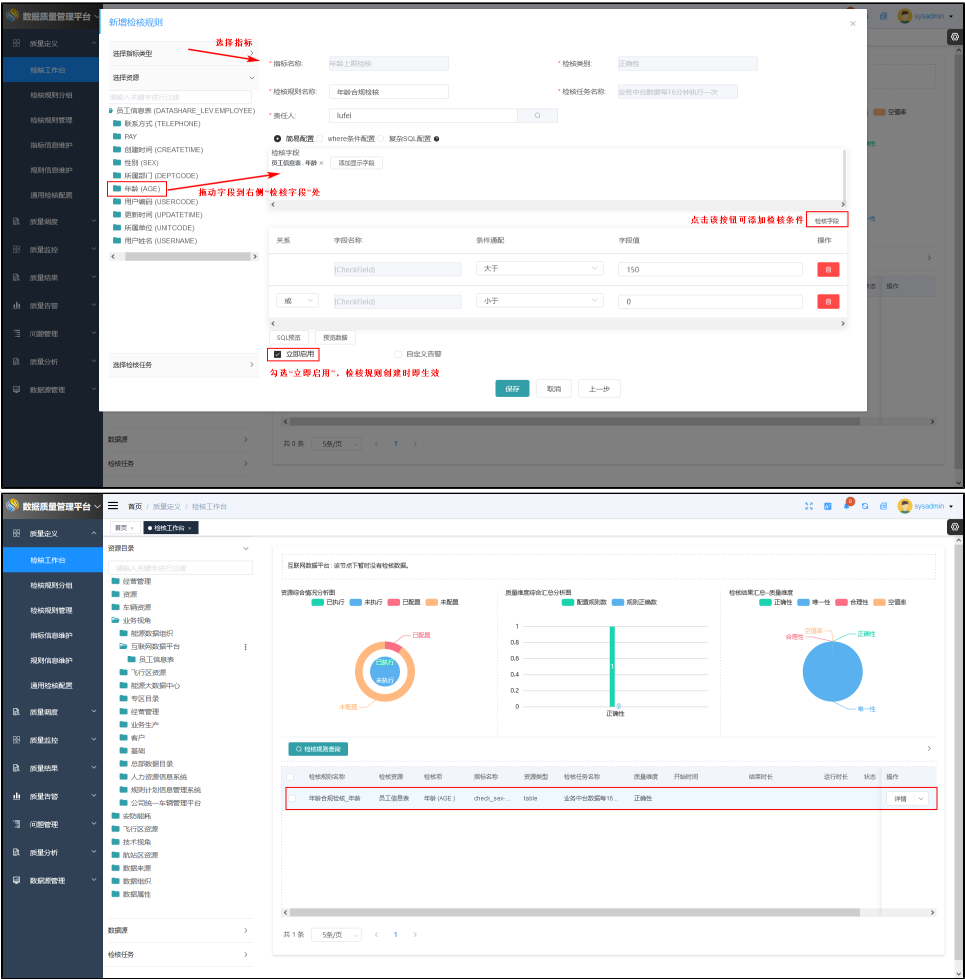
2. 点击“新增检核规则”按钮，选择要检核的类型，本示例以DB关系型数据库检核为例如下图。



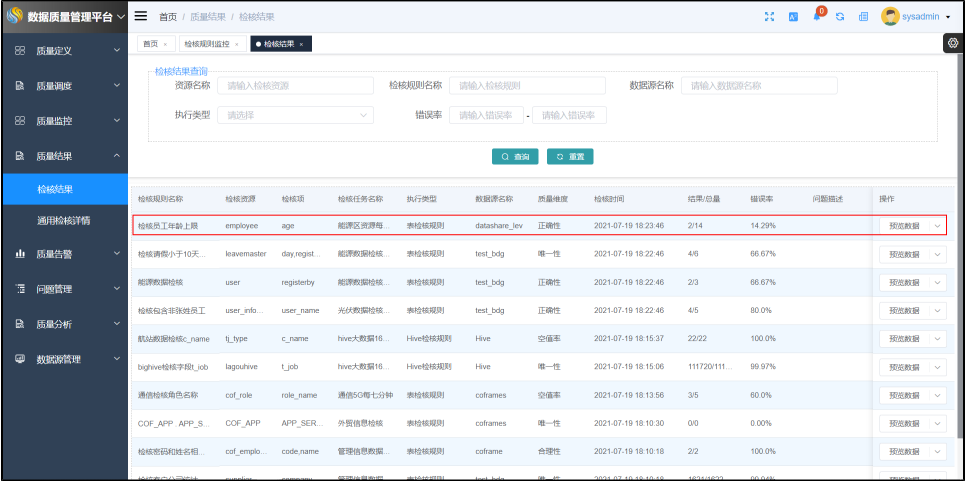
3. 在“新增检核规则”页面，选择指标及检核任务，拖动资源页面的字段至右侧“检核字段”处，添加检核条件，保存后检核规则创建成功。

说明

检核指标、检核任务已创建，创建步骤可参考：[指标信息维护](#)、[检核任务管理](#)。



4. 点击“质量结果>检核结果”页面，可查看检核规则执行的结果，对结果进行预览、下载。



3.4 数据质量检核

参考[3.1 数据质量检核快速入门](#)。

3.5 服务发布

服务发布流程可参考：[资产服务发布快速入门](#)。

3.6 数据标准快速入门

参考[3.1 标准采集快速入门](#)。

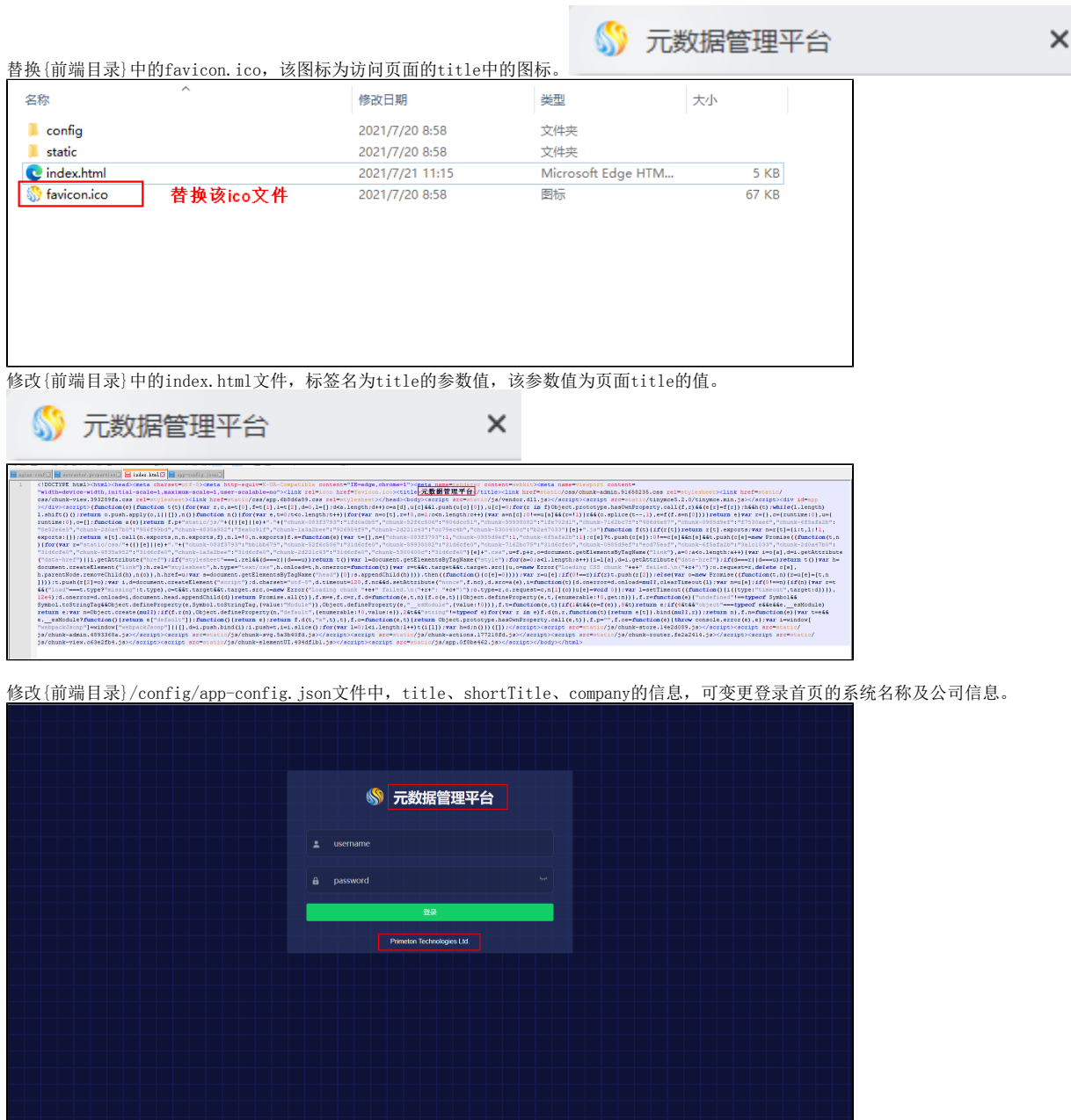
04 前端显示配置

因项目需要或其他原因，需要修改前端页面显示的系统名称及LOGO等，可参考如下方式进行配置。此处以元数据管理平台修改为示例，其余系统配置方式相同。

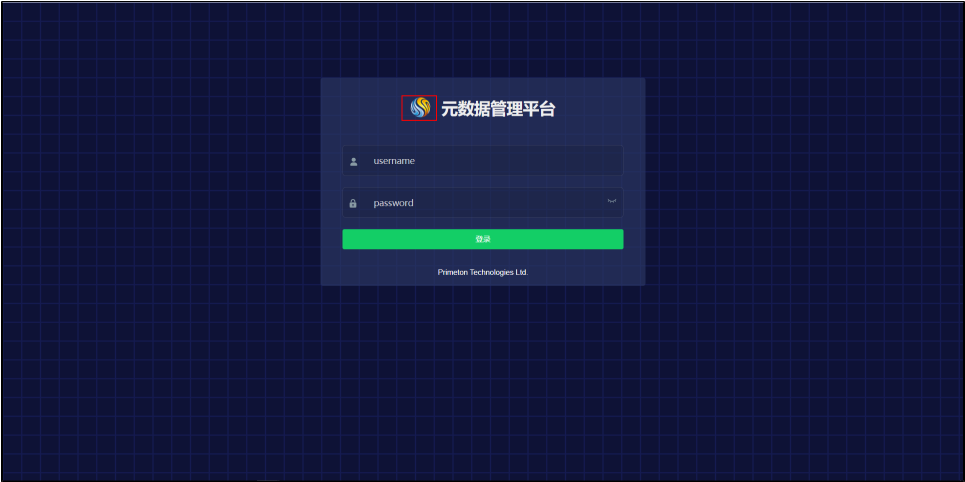
注意

替换使用的图片需和原图片格式、像素一致，否则可能会出现显示不全或显示异常。

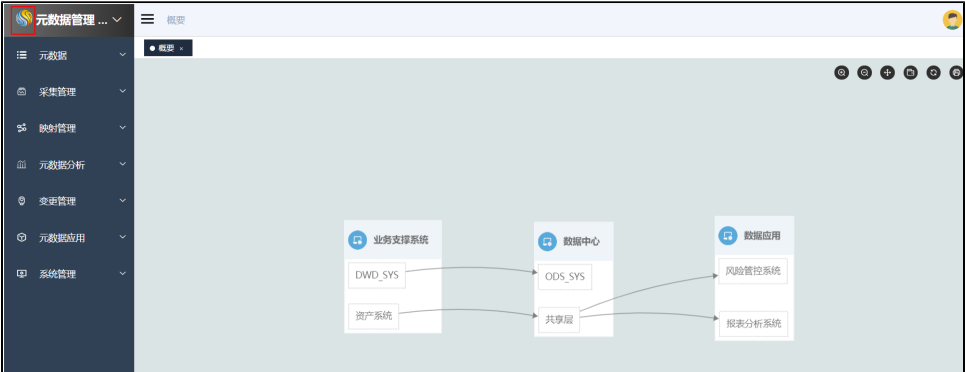
修改完后需要刷新前端展示，使修改立即生效，使用`nginx -s reload`命令。



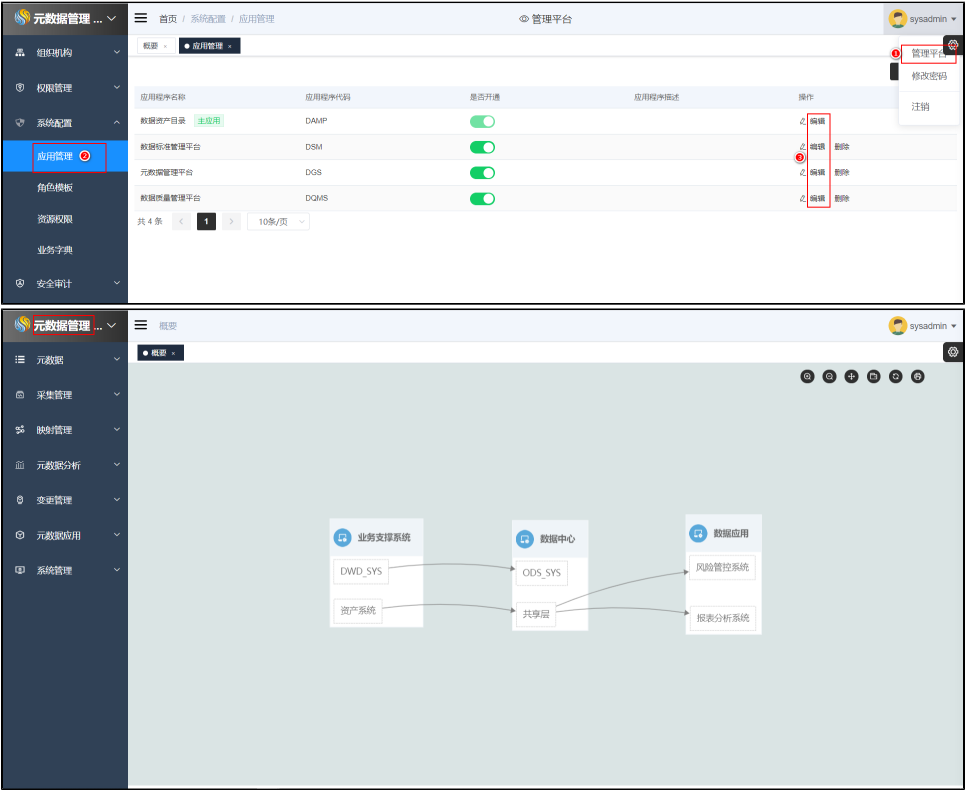
4. 替换(前端目录)/static/img目录下的logo.925bd014.png图片文件，该LOGO为首页页面左上角显示的LOGO。



5. 替换(前端目录)/static/images目录下的logo.png图片文件，该LOGO为登录后页面左上角显示的LOGO。



6. 使用管理员登录系统后，点击右下角下拉菜单的“管理平台”链接后，点击“系统配置>应用管理”，在应用程序列表中点击编辑可修改应用程序名称。该值为登录后页面左上角的应用名称。



说明

集成环境中，分别在不同应用的前端包中修改以上内容即可对集成中所有应用LOGO等配置进行替换。

05 用户操作手册

集成应用的用户操作手册，请参考各应用的操作手册，链接如下：

元数据管理平台操作手册：[04 用户操作手册](#)

数据标准管理平台操作手册：[04 用户操作手册](#)

数据资产目录操作手册：[04 用户操作手册](#)

数据质量管理平台操作手册：[04 用户操作手册](#)

06 FAQ

1. 元数据或数据标准server启动报错：Error creating bean with name 'dgsRemoteRpc' defined in class path resource [spring/corebean/context-adapter-mm-server.xml]

当启动元数据或数据标准Server应用时出现如下报错：

```
2021-09-01 15:56:53.351 [main] ERROR org.springframework.boot.SpringApplication - Application run failed
org.springframework.beans.factory.BeanCreationException: Error creating bean with name 'dgsRemoteRpc' defined in class path resource [spring/corebean/context-adapter-mm-server.xml]: Invocation of init method failed; nested exception is
java.io.IOException: Failed to bind
    at org.springframework.beans.factory.support.AbstractAutowireCapableBeanFactory.initializeBean(AbstractAutowireCapableBeanFactory.java:1745)
    at org.springframework.beans.factory.support.AbstractAutowireCapableBeanFactory.doCreateBean(AbstractAutowireCapableBeanFactory.java:576)
    at org.springframework.beans.factory.support.AbstractAutowireCapableBeanFactory.createBean(AbstractAutowireCapableBeanFactory.java:498)
    at org.springframework.beans.factory.support.AbstractBeanFactory.lambda$doGetBean$0(AbstractBeanFactory.java:320)
    at org.springframework.beans.factory.support.DefaultSingletonBeanRegistry.getSingleton(DefaultSingletonBeanRegistry.java:222)
    at org.springframework.beans.factory.support.AbstractBeanFactory.doGetBean(AbstractBeanFactory.java:318)
    at org.springframework.beans.factory.support.AbstractBeanFactory.getBean(AbstractBeanFactory.java:199)
    at org.springframework.beans.factory.support.DefaultListableBeanFactory.preInstantiateSingletons(DefaultListableBeanFactory.java:846)
    at org.springframework.context.support.AbstractApplicationContext.finishBeanFactoryInitialization(AbstractApplicationContext.java:863)
    at org.springframework.context.support.AbstractApplicationContext.refresh(AbstractApplicationContext.java:546)
    at org.springframework.boot.web.servlet.context.ServletWebServerApplicationContext.refresh(ServletWebServerApplicationContext.java:142)
    at org.springframework.boot.SpringApplication.refresh(SpringApplication.java:775)
    at org.springframework.boot.SpringApplication.refreshContext(SpringApplication.java:397)
    at org.springframework.boot.SpringApplication.run(SpringApplication.java:316)
    at com.primeton.standard.boot.Application.main(Application.java:134)
Caused by: java.io.IOException: Failed to bind
    at io.grpc.netty.NettyServer.start(NettyServer.java:256)
    at io.grpc.internal.ServerImpl.start(ServerImpl.java:167)
    at io.grpc.internal.ServerImpl.start(ServerImpl.java:91)
    at com.primeton.dgs.kernel.common.remote.GrpcEngine.reconnect(GrpcEngine.java:119)
    at com.primeton.dgs.kernel.common.remote.GrpcEngine.afterPropertiesSet(GrpcEngine.java:85)
    at org.springframework.beans.factory.support.AbstractAutowireCapableBeanFactory.invokeInitMethods(AbstractAutowireCapableBeanFactory.java:1804)
    at org.springframework.beans.factory.support.AbstractAutowireCapableBeanFactory.initializeBean(AbstractAutowireCapableBeanFactory.java:1741)
    ... 14 common frames omitted
Caused by: java.net.BindException: 无法指定被请求的地址
    at sun.nio.ch.Net.bind0(Native Method)
    at sun.nio.ch.Net.bind(Net.java:461)
    at sun.nio.ch.Net.bind(Net.java:453)
    at sun.nio.ch.ServerSocketChannelImpl.bind(ServerSocketChannelImpl.java:222)
    at io.netty.channel.socket.nio.NioServerSocketChannel.doBind(NioServerSocketChannel.java:130)
    at io.netty.channel.AbstractChannel$AbstractUnsafe.bind(AbstractChannel.java:558)
    at io.netty.channel.DefaultChannelPipeline$HeadContext.bind(DefaultChannelPipeline.java:1358)
    at io.netty.channel.AbstractChannelHandlerContext.invokeBind(AbstractChannelHandlerContext.java:501)
    at io.netty.channel.AbstractChannelHandlerContext.bind(AbstractChannelHandlerContext.java:486)
    at io.netty.channel.DefaultChannelPipeline.bind(DefaultChannelPipeline.java:1019)
    at io.netty.channel.AbstractChannel.bind(AbstractChannel.java:254)
    at io.netty.bootstrap.Bootstrap$2.run(Bootstrap.java:366)
    at io.netty.util.concurrent.AbstractEventExecutor.safeExecute(AbstractEventExecutor.java:163)
    at io.netty.util.concurrent.SingleThreadEventExecutor.runAllTasks(SingleThreadEventExecutor.java:404)
    at io.netty.channel.nio.NioEventLoop.run(NioEventLoop.java:466)
    at io.netty.util.concurrent.SingleThreadEventExecutor$5.run(SingleThreadEventExecutor.java:897)
    at io.netty.util.concurrent.FastThreadLocalRunnable.run(FastThreadLocalRunnable.java:30)
    at java.lang.Thread.run(Thread.java:748)
```

问题原因及解决办法：

主要是由于rmi监听端口被占用，需要查看t_sys_paramitem表中的以下端口是否已经被占用，若端口被占用，则修改为未被占用的端口。

对象 无标题 - 查询 t_sys_paramitem @product_integrat...

开始事务 文本 筛选 排序 导入 导出

☒ ITEM_CODE 包含 rmi

应用

ITEM_CODE	PARAM_CODE	ITEM_VALUE	ITEM_ORDER	DESCRIPTION
DSM.rmi.web.host	DSM.Extractor-Server-Setting	192.168.16.166	3	Web端
DSM.rmi.web.port	DSM.Extractor-Server-Setting	9996	4	Web端
rmi.web.host	Extractor-Server-Setting	192.168.16.166	3	Web端
rmi.web.port	Extractor-Server-Setting	9998	4	Web端